rechentechnik
datenverarbeitung

rd

3·89
26. Jahrgang
März 1989

DDR · 3,40 M
Verlag
Die Wirtschaft

SOFTWARE-TECHNOLOGIE

260296

INHALT

Softwaretechnologie

5

Dr. Bernd Goes: Anforderungen an die Implementierung von Software	Д-р Бернд Гёс: Требования к имплементации матобеспечения	Dr. Bernd Goes: Demands made on the implementation of software

6

Wolfram Lüneburg, Günter Reinemann, Wolfram Weiß: Schnittstellengestaltung komplexer CAE-Software	Вольфрам Люнебург, Гюнтер Райнеман, Вольфрам Вайс: Оформление интерфейса комплексного матобеспечения САИТ	Wolfram Lüneburg, Günter Reinemann, Wolfram Weiss: Design of interfaces of complex CAE software

9

Frank Jauch: Effektive Softwareentwicklung mittels Trennung von Programm- und Datenstruktur	Франк Яух: Эффективная разработка матобеспечения путем разделения структуры программ и данных	Frank Jauch: Effective software development by separation of program and data structure

10

Dr. Eberhard Börner: EXPRESS – Entwicklungs- und Steuersystem für REDABAS-Anwendungen auf 16-Bit-PC	Д-р Эберхард Бёрнер: ЭКСПРЕСС – Система разработки и применения РЕДАБАЗ на 16-разрядных ПК	Dr. Eberhard Börner: EXPRESS – Development and control system for REDABAS applications on 16-bit PC

13

Bernd Blacha: MGEN – ein Maskengenerator	Бернд Блаха: Генератор масок МГЕН	Bernd Blacha: MGEN – A mask generator

17

Rudolf Klar: UDIS – Universelles Dialogsystem	Рудольф Клар: Универсальная диалоговая система УДИС	Rudolf Klar: UDIS – Universal dialogue system

19

Dr. Eckhart Richter: LS2 – Softwaremonitor und Steuersystem für SVM	Д-р Экхарт Рихтер: ЛС 2 – монитор матобеспечения и система управления для SVM	Dr. Eckhart Richter: LS2 – Software monitor and control system for SVM

23

Claus Dieter Krah, Roland Baum, Hartmut Schettler, Klaus Ginter: Dialogorientiertes Auskunftssystem für ESER	Клаус Дитер Кра, Роланд Баум, Хартмут Шеттлер, Клаус Гинтер: Ориентированная на диалог справочная система ЕС ЭВМ	Claus Dieter Krah, Roland Baum, Hartmut Schettler, Klaus Ginter: Dialogue-oriented information system for use with ESER (UCS)

24

Dr. M. Feder, Dr. G. Klatt: Nutzerbezogene Bewertung automatisierter Informationssysteme	Д-р Манфред Федер, Д-р Фюнтер Клатт: Оценка автоматизированных информационных систем с ориентацией на пользователя	Dr. M. Feder, Dr. G. Klatt: User-oriented evaluation of automated information systems

27

Jan-Markus Trauzettel: Dialogorientiertes Demonstrationsprogramm zur Wissensvermittlung	Ян-Маркус Трауцеттель: Ориентированная на диалог демонстрационная программа передачи знаний	Jan-Markus Trauzettel: Dialogue-oriented demonstration program for imparting knowledge

28

Roland Garz: Interaktive 3D-Modellierung	Роланд Гарц: Интерактивное трехразмерное моделирование	Roland Garz: Interactive 3D-modelling

30

Frank Gebhardt: PSU und CCS für DOS-4/ES	Франк Гебхардт: PSU и CCS для ДОС-4/ЕС	Frank Gebhardt: PSU (parallel-serial converter) and CCS for DOS-4/ES

Verlag Die Wirtschaft Berlin
Am Friedrichshain 22, Berlin, 1055,
Ruf 4 38 70, Fernschreiber 114566

Verlagsdirektor: Dieter Grüneberg

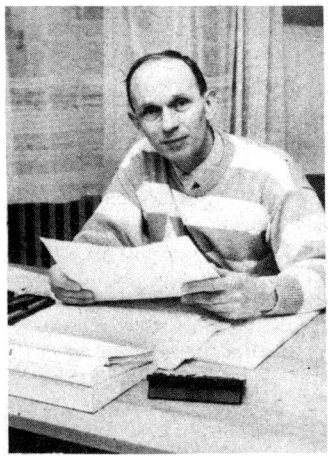
Redaktion rechentechnik/datenverarbeitung
Am Friedrichshain 22, Berlin, 1055,
Chefredakteur: Franz Loll 4 38 73 41
Redakteure:
Klaus Fischer 4 38 73 38
Monika Schalwat 4 38 73 53
Claudia Schulz 4 38 73 16

Sekretariat: 4 38 72 33
Titelgestaltung: Rainer Dassow
Lizenz dés Presseamtes beim Vorsitzenden des Ministerrates der DDR. Nr. 1529 · EDV-Art.-Nr. 83517
Redaktionsschluß: 25. 1. 1989
rechentechnik/datenverarbeitung erscheint monatlich zum Bezugspreis DDR 3,40 M je Heft.
Auslandspreise sind dem Zeitschriftenkatalog des Außenhandelsbetriebes Buchexport zu entnehmen.
Satz: Verlag Die Wirtschaft
Druck: (140) Neues Deutschland
Anzeigenverwaltung: Berliner Verlag, Karl-Liebknecht-Str. 29 Berlin, 1056 Telefon: 2 70 33 02
Anzeigenannahme: Berliner Verlag und Annahmestellen in Berlin und in den Bezirken. Zur Zeit gültige Anzeigenpreisliste Nr. 12
Im Ausland: INTERWERBUNG GmbH – Gesellschaft für Werbung und Auslandsmessen der DDR, Hermann-Duncker-Str. 89 Berlin, 1157, DDR
Bestellungen aus der DDR sind an den Postzeitungsvertrieb zu richten Inkasso-Zeitraum: monatl.
Im Ausland
In den sozialistischen Ländern nur der zuständige Postzeitungsvertrieb. In allen anderen Staaten der örtliche Buch- und Zeitschriftenhandel. Bestellungen des Buch- und Zeitschriftenhandels sind zu richten an BUCHEXPORT Volkseigener Außenhandelsbetrieb der DDR, Leninstr. 16, Postfach 160, DDR – Leipzig, 7010 oder an Verlag Die Wirtschaft Berlin, Am Friedrichshain 22, Berlin, 1055, DDR

Vervielfältigungen dieser Ausgabe oder rechtlich geschützter Beiträge nur mit Genehmigung des Rechtsinhabers und Quellenangabe

Mitglieder des Redaktionsbeirates
Dr. sc. Claus Goedecke · Dr. Rolf Gräßler · Prof. Dr. sc. Gerhard Keßler · Dr. Rolf Kilian · Hans Kunau · Walter Münch Axel Rathsack · Prof. Dr. sc. Gerd Rossa · Prof. Dr. sc. Claus Sattler · Prof. Dr. sc. Wolfgang Schoppan (Vorsitzender) · Dr. Werner Schulze · Horst Stoll · Prof. Dr. Franz Stuchlik · Dr. Dieter Urban

Softwaretechnologische Konzepte und Orientierungen



Dr. Claus Goedecke

Die Softwaretechnologie zielt auf eine effektive Entwicklung, Anwendung und Wartung qualitativ hochwertiger Softwareprodukte. Entsprechende softwaretechnologische Entwicklungsarbeiten werden in der DDR bereits seit einer Reihe von Jahren durchgeführt. Im Ergebnis entstanden unterschiedliche multivalent nutzbare Mittel und Systeme zur Herstellung von Software, mit denen eine systematisierte, methoden- und rechnergestützte Softwareentwicklung unterstützt wird.

Ausgangspunkt einer durch die Softwaretechnologie begründeten zielgerichteten, ingenieurmäßigen Vorgehensweise in der Softwareentwicklung bildet das Phasenkonzept gemäß TGL 44545 „Phasenmodell des Softwarelebenszyklus", durch das eine Strukturierung des Entwicklungsablaufes vorgegeben wird. Darauf aufbauend werden die inhaltlichen Aspekte der Softwaretechnologie bestimmt durch

– den Einsatz moderner Prinzipien und Methoden (Methodenkonzept),
– die Nutzung von Werkzeugen und vorgefertigten Programmkomponenten (Werkzeugkonzept) sowie in unmittelbarer Wechselwirkung ergänzt durch
– die Anwendung leistungsfähiger Programmiersprachen (Sprachkonzept).

Für die genannten Konzepte bestehen in der DDR folgende Vorgaben:

Methodenkonzept

Korrespondierend mit dem Phasenkonzept bildet das „Handbuch der Softwaretechnologie Robotron", das vom VEB Robotron-Projekt Dresden 1984 bereitgestellt wurde, die Arbeitsgrundlage der Softwaretechnologie in der DDR. Ergänzend dazu stehen u. a. zur Verfügung:

– TGL 44530: Termini und Definitionen
– TGL 44535: Dokumentation von Programmen, Programmsystemen und Software-Anwendungssystemen.

In den Bereichen der Volkswirtschaft existieren darüber hinaus eine Reihe zusätzlicher Materialien, die das Methodenkonzept weiter untersetzen.

Werkzeugkonzept

Als Grundlage für die Entwicklung und Anwendung von Softwarewerkzeugen wurde 1988 eine Neufassung des „Orientierungsmaterials über technologische Mittel und Systeme der Kombinate Robotron und Datenverarbeitung" erarbeitet, das neben der Nutzung in beiden Kombinaten allen Anwendern in der DDR einen Überblick über verfügbare technologische Mittel und Systeme vermittelt. Entscheidend kommt es bei diesen Entwicklungen darauf an, daß für die Endnutzer die Entwicklungsumgebungen für unterschiedliche Rechnerklassen (ESER-, 32- und 16-Bit-Technik) weitgehend einheitlich gestaltet werden. Zu den Werkzeugen gehören u. a.:

– Technologisches System zur Softwareentwicklung TESYS-3, das – unter Berücksichtigung der Weiterentwicklung zu TESYS-4 – für alle Rechnerklassen einsetzbar ist und die Zielsprachen COBOL, PL/1, C und FORTRAN 77 unterstützt
– Programmierumgebung für MODULA-2
– Projektierungs- und programmierungsstützende Umgebung PSU für ESER-Betriebssysteme als Entsprechung der MUTOS-Systeme für 32- und 16-Bit-Technik
– Entwurfs- und Implementierungswerkzeuge des VE KDV, u. a. Datenstrukturbibliothek, Quelltextstrukturierung, Job-Control-Syntaxchecker, Maskengenerator und Universelles Dialogsystem, die ebenfalls für unterschiedliche Rechnerklassen zur Verfügung stehen. Die beiden letztgenannten Komponenten werden im Schwerpunktkomplex des vorliegenden Heftes vorgestellt.

Außerdem existieren in der DDR eine Vielzahl weiterer spezieller Komponenten, die unter anderem auch in den rd-Rubriken *Softwaremarkt* und *Edv-Ratioforum* publiziert werden.

Sprachkonzept

Grundlage für die Anwendung universeller Programmiersprachen bildet die „Richtlinie zur einheitlichen Anwendung von Betriebssystemen, universellen Programmiersprachen und Datenbanksoftware in der DDR", die 1988 vom VEB Kombinat Robotron unter anderem in Abstimmung mit der Akademie der Wissenschaften und dem VE Kombinat Datenverarbeitung herausgegeben wurde. Unter Beachtung des Portabilitätsaspektes ist dabei vorrangig auf die Anwendung solcher Sprachen zu orientieren, die durchgängig über verschiedene Rechnerklassen und auf weitgehend einheitlichem Sprachniveau verfügbar sind. Dazu gehören insbesondere MODULA-2, C und FORTRAN 77.

Software für Dienstleistungen

(rd-Sta) Im Zusammenhang mit der Aufnahme der Tätigkeit eines Leitzentrums zur Applikation von/zur Information über CAD/CAM-Lösungen für den Bereich der Örtlichen Versorgungswirtschaft (ÖVW) wird die Softwareerarbeitung und -nutzung durch das Institut für Kommunalwirtschaft Dresden koordiniert.

Anläßlich der Tage der Rationalisierung der ÖVW wurde durch das Leitzentrum CAD/CAM eine erste Softwareübersicht mit 283 nachnutzungsfähigen Programmen rechnergestützt aufbereitet und den Interessenten zur Verfügung gestellt.

Dazu gehören u. a.

46 Softwarelösungen für hauswirtschaftliche Dienstleistungsbetriebe, 45 Lösungen für stadtwirtschaftliche Dienstleistungsbetriebe, 11 Lösungen für Textilreinigungsbetriebe und 153 Programmlösungen für Produktionsgenossenschaften des Handwerks (PGH), Einkaufs- und Liefergenossenschaften des Handwerks (ELG) und private Handwerker.

Folgende Aufgabengebiete bzw. -bereiche stellen dabei die Schwerpunkte der Softwarelösungen dar:

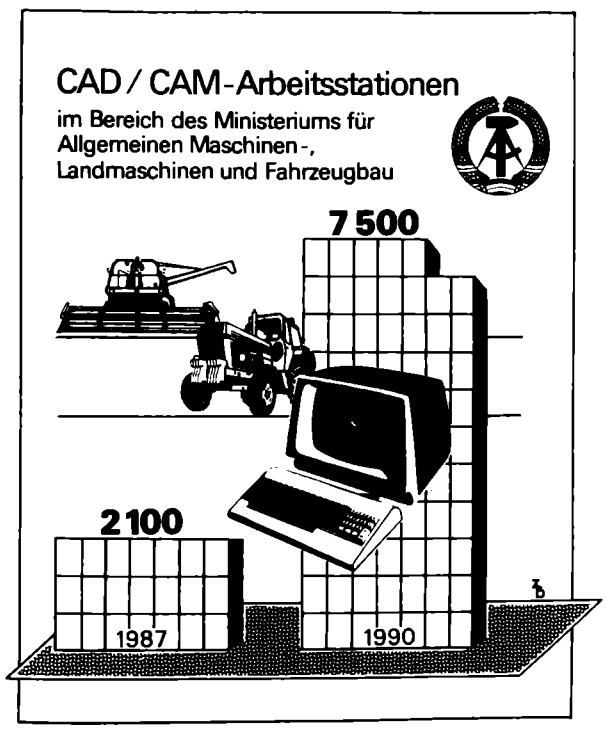
– technologische Planung, Vorbereitung und Organisation der leistungsdurchführenden Prozesse in den VE Dienstleistungsbetrieben (DLB), PGH und ELG, wie Maßanfertigung von Bekleidung, Weißnähen, Glas- und Gebäudereinigung;

– Bearbeitung der Kundendienstprozesse, des Vertriebs sowie der Transport-, Umschlags- und Lagerprozesse in den VE DLB, PGH und ELG, wie rechnergestützte Arbeitsplätze in Komplexannahmestellen und Auftragszentralen Dienstleistungen der DLB;

– Bearbeitung von Aufgaben in der Planung, Rechnungsführung und Statistik in den VE DLB, PGH und ELG und in privaten Handwerksbetrieben, wie die komplexe Planung und Abrechnung eines VE DLB, die Rechnungslegung für Haushaltfertigwäsche der Textilreinigung, die Kosten-, Erlös- und Finanzrechnung für die PGH, die Einnahme- und Ausgaberechnung sowie Lohn- und Gehaltsrechnung für private Handwerksbetriebe.

In Zusammenarbeit mit den VEB Rechnungsführung und Wirtschaftsberatung der Bezirke, den VE DLB der ÖVW, den Fachorganen ÖVW der Räte der Bezirke und den Handwerkskammern der Bezirke erfolgt die ständige Ergänzung und Aktualisierung dieses Softwarespeichers, der den o. g. Einrichtungen persönlich zur Verfügung gestellt wird. Interessenten wenden sich deshalb an die territorial zuständigen Verantwortlichen für Rechentechnik der o. g. Einrichtungen oder das Institut für Kommunalwirtschaft Dresden, Leitzentrum CAD/CAM der ÖVW.

(rd-ADN/ZB) Mit der Erhöhung der CAD/CAM-Arbeitsstationen auf 7500 im Jahr 1990 sollen im Gesamtbereich des Ministeriums über 14000 Arbeitskräfte gewonnen werden.



China intensiviert Computerproduktion

(rd-ADN) Die VR China plant in den nächsten Jahren eine Steigerung der Jahresproduktion von Computern um nahezu das Dreifache. Bis 1990 sollen jährlich Rechner im Werte von sechs Milliarden Yuan hergestellt werden. Wie die chinesische Nachrichtenagentur Hsinhua meldete, verzeichnete die Computerindustrie von 1981–1987 durchschnittlich einen Zuwachs von 30 Prozent pro Jahr. 1987 wurden Computer mit einem Wert von 2,1 Milliarden Yuan hergestellt. Obwohl China gegenwärtig mittlere und Großcomputer – darunter solche zur Verarbeitung chinesischer Schriftzeichen – in mehreren Varianten herstellen könne, sei die internationale Wettbewerbsfähigkeit weiterhin gering. Der Ausbau des Industriezweiges soll auch zu einer Qualitätssteigerung in diesem Bereich beitragen und den Absatz auf Überseemärkten fördern.

Technische Diagnostik

(rd-ADN-ZB) Mittels rechnergestützter technischer Diagnostik erfolgt im Kombinat Agrochemie Piesteritz die zustandsorientierte Instandsetzung von Elektromotoren in chemischen Produktionsanlagen. Transportable Schwingungsdiagnosegeräte liefern in Verbindung mit einem Kleinrechner genaue Daten über Abnutzung und Verschleiß sowie die verbleibende Restnutzungsdauer des Motors. Instandhaltungsmaßnahmen werden so für die produktiven Abteilungen planbar. Heiko Otto (Foto oben) arbeitet an einem Diagnoseplatz, der auch eine grafische Darstellung bei der Fehlersuche ermöglicht. Die Diagnosestation der Elektrowerkstatt – eine von derzeit 127 CAD/CAM-Stationen des Kombinates – wird 14 Stunden pro Tag ausgelastet.

Weiterbildungsangebot auf dem Gebiet Informatik

Nach dem erfolgreichen Abschluß des Studienjahres 1987/88 legt das Weiterbildungszentrum Informatik am Organisations- und Rechenzentrum der Humboldt-Universität ein neues Programm vor. Es folgt dem bisherigen Trend einer anspruchsvollen Weiterbildung für Informatiker und Interessenten angrenzender Disziplinen.

Bei der Konzipierung des neuen Programms wurde Wert auf eine umfassende und gründliche Be-

handlung der Themen gelegt. Damit mußte – auch aus Kapazitätsgründen – auf eine größer denkbare Vielfalt verzichtet werden.
Schwerpunktmäßig wurden bei der Gestaltung des neuen Bildungsangebotes Künstliche Intelligenz- und Rechnernetz-Themen ausgebaut. Damit soll ein Beitrag zum Bildungsvorlauf auf jenen Gebieten der Informatik geleistet werden, die einer rasanten Entwicklung unterliegen. Es geht um die Vermittlung wissenschaftlicher Ergebnisse, die Praxisreife erlangt haben, aber von kommerziellen Schulungseinrichtungen noch nicht oder eingeschränkt berücksichtigt werden.
Um das eigene Angebot abzurunden, wurden auch einige herkömmliche Themen aufgenommen.
Folgende Themen stehen zur Auswahl:
– Grundlagen von LAN, Leistungen, DDR-Lösungen
– Anwendung der Rechnernetz-Software SM-Net
– Lichtwellenleitereinsatz in lokalen Rechnernetzen
– Entwicklung von Kommunikationsprotokollen
– Management in verteilten Systemen
– Bürokommunikation , – kommunikationstechnologische Grundlagen, Dienste, Anwendungen
– Wissensrepräsentation und -verarbeitung
– Programmierung in GC-LISP
– Programmierung mit Prolog am Beispiel von S-PROLOG
– Expertensysteme
– Schlußfolgerungsverfahren
– Probleme der Gestaltung von Nutzerschulungen und der Wissenserfassung für Expertensysteme
– Digitale Grafik und Bildanalyse
– Assemblerprogrammierung unter MS-DOS
– VMX – UNIX auf ESER
– Programmiersprache C
– Statistikprogrammpakete auf Mikrorechnern
– Simulation mittels Computer
– Numerik gewöhnlicher Differentialgleichungen und Simulation kontinuierlicher Systeme auf PC/XT und /AT.
Nähere Einzelheiten können dem Weiterbildungsangebot entnommen werden, das auf Anfrage zugestellt wird.
Interessenten wenden sich bitte an die Humboldt-Universität zu Berlin, ORZ, Weiterbildungszentrum Informatik, PSF 1297, Berlin 1086, Tel. 20 93 23 21.

Computerverbindung zwischen der UdSSR und den USA

(rd-TASS) Eine Computer-Fernverbindung zwischen der UdSSR und den USA ist erstmals von der USA-Gesellschaft „San-Fransisco-Moscow-Teleport" und dem Forschungsinstitut für angewandte automatisierte Systeme – der Dachorganisation der UdSSR für die automatische Verbindung mit Computernetzen in anderen Ländern – hergestellt worden.
Bei der Probe-Nachrichtenübertragung wurde der internationale Nachrichtensatellit „Intelsat" als Zwischensender eingesetzt. Die Nachauftragnehmer sind: von sowjetischer Seite das Ministerium für Fernmeldewesen der UdSSR und von amerikanischer Seite die Gesellschaft I. D. G. Communication Group.
„Wir können uns an die Computernetze aller sozialistischen Länder sowie Österreichs und Finnlands leicht anschließen", sagte der Direktor des Instituts für angewandte automatisierte Systeme, Oleg Smirnow, gegenüber der Zeitung Sowjetskaja Rossija. Zugleich verwies er darauf, daß die Übertragung von Nachrichten in die USA durch Vermittlung anderer Länder erfolgt und daß die UdSSR hohe Summen dafür zahlen muß. Um die Verbindungsmöglichkeiten für Vertretungen verschiedener Gemeinschaftsunternehmen, Nichtregierungsorganisationen oder Betriebe zu gewährleisten, die mit ausländischen Partnern zusammenarbeiten, wurde beschlossen, gemeinsam mit „San-Fransisco-Moscow-Teleport" eine gemeinsame Firma zu gründen und Dienstleistungen bei der Computer-Fernverbindung zu gewähren. Nach Ansicht von Smirnow wird der neue Nachrichtenkanal es gestatten, den Austausch von Informationen zwischen der UdSSR und den USA wesentlich zu vergrößern.

Internationaler Computerklub in Moskau gegründet

(rd-TASS) Internationaler Computerklub heißt eine internationale nichtstaatliche Organisation, die im Dezember 1988 in Moskau gegründet wurde. Sie soll dazu beitragen, die Ausstattung der verschiedenen Bereiche der Volkswirtschaft der UdSSR mit Rechentechnik zu beschleunigen, die Bevölkerung besser mit der Computertechnik vertraut zu machen und den internationalen Austausch auf diesem Gebiet zu erweitern. Zu den Gründern des Klubs gehören die Industrie- und Handelskammer der UdSSR, die Akademie der Wissenschaften der UdSSR, das Büro des Ministerrates der UdSSR für Maschinenbau, die Unionsgesellschaft „Snanije" sowie hunderte sowjetischer Industriebereiche. Unter den ausländischen Partnern befinden sich eine Reihe von Firmen, die an Lieferungen von Rechentechnik an die UdSSR interessiert sind.
Zu den Aufgaben des Klubs gehört die Entwicklung der internationalen Zusammenarbeit auf dem Gebiet der Informatik, der Rechentechnik, der Schaffung von Datenbanken und des Austausches von Programmen. Eine wichtige Richtung in der Arbeit des Klubs ist auch der Schutz des Urheberrechts der sowjetischen Programmierer.
Das Grundkapital des Klubs, der nicht kommerziellen Charakter hat, beträgt 500 000 Rubel. Es ist vorgesehen, daß interessierte Mitglieder des Klubs Sonderfonds für Zielprogramme schaffen können. In der Perspektive sollen auch in anderen Regionen der UdSSR Computerzentren entstehen.

Computer der fünften Generation sind nicht zu realisieren

(rd-ADN) Fehlgeschlagen scheinen die ehrgeizigen Pläne der japanischen Regierung, mit den Computern der fünften Generation einen völlig neuen Typ von Rechenmaschinen zu konstruieren. Auf einer Konferenz im Dezember in Tokyo mußten die Initiatoren des Programms zugeben, daß ihre ursprünglichen Ziele nicht zu realisieren seien.
Kazuhiro Fuchi, Chef des Icot (Institut für neue Computer-Technologien) und eine Schlüsselfigur in diesem Forschungsprogramm, erklärte, daß er nicht mehr mit dem Bau eines Computer-Prototyps der fünften Generation rechne.
450 Mio. Dollar wollte das japanische Handelsministerium (Miti) bis 1992 in das Programm investieren. Fuchi, dessen Institut davon den Hauptanteil erhält, will sich jetzt verstärkt den neuronalen Netzwerken und der Parallelverarbeitung zuwenden.

PC-MOS/386 als Mehrbenutzeralternative zu MS-DOS

Die fehlenden Multiuser- und Multitasking-Fähigkeiten von MS-DOS lassen die Software-Entwickler nicht ruhen. Unermüdlich suchen sie nach Möglichkeiten, den biederen PC auch dieses Leistungsspektrum zu eröffnen. Eine Lösung in dieser Richtung kommt aus den USA: PC-MOS/386, ein Betriebssystem, das für alle Rechnerklassen vom XT bis zum 32-Bit-Rechner (80386er) bestimmt ist.
Bei diesem Betriebssystem können die Benutzer, die selbsredend vorher auf der DOS-Ebene gearbeitet haben, ihre bisherigen Programme und Applikationen laut Angaben von Software Link, dem Hersteller des Systems, weiterhin benutzen. Nur mit dem bedeutsamen Unterschied, daß dies nun auch in Mehrbenutzerumgebung möglich ist, verbunden mit den Vorzügen des Multitasking.
Bei der jüngst vorgestellten Version von PC-MOS/386 ist besonders erfreulich zu vermerken, daß sie auch CGA-Grafik unterstützt. Das bedeutet für alle Anwender, die etwa Lotus 1-2-3, Autocad oder vergleichbare Programme im Colour-Graphic-Mode verwenden, daß sie nun diese Programme auf gewohnte Art auf dem Grafik-Terminal nutzen können. PC-MOS/386 ersetzt, wie Compucon, der BRD-Distributor, erklärt, das bisherige Betriebssystem DOS vollständig. Dabei ist der Grundbefehlssatz weitgehend identisch mit den gewohnten DOS-Befehlen; allerdings wurde er für die speziellen Mehrbenutzer- und Multitasking-Fähigkeiten um einige Befehle erweitert.

Bildungscomputer im Test

Ende November des vergangenen Jahres fand in Dresden eine Anwendertagung zum Bildungscomputer statt. Bis zum Jahresende gingen weitere Berichte ein, die einen guten Überblick über die Leistungsfähigkeit der ausgelieferten Test-Geräte geben.

Die Testpartner vom Computerkabinett des Polytechnischen Zentrums des VEB Berliner TT-Bahnen, von Hochschulen und Einrichtungen der Berufsbildung haben mit sichtlicher Begeisterung den „Neuen" in Besitz genommen.

Was offenbar den meisten Spaß gemacht hat, sind die zuverlässig funktionierende Tastatur, die zahlreichen Möglichkeiten der Bildschirm- und Speicherarbeit und die vielen Grafik-Varianten. Auch mit der Diskettenarbeit gab es keinerlei Probleme. Einmal verwöhnt von der guten Gestaltung der Unterlagen zum KC 87, sind die Anwender punktbezogen kritisch auf das Programmierhandbuch eingegangen, welches den Verfassern wertvolle Hinweise gegeben hat. Auch für die Belegung der Tastatur und ihre Beschriftung gab es Wünsche, die aber nicht alle berücksichtigt werden können, weil sie sich naturgemäß teilweise widersprechen.

Die Erstellung der BASIC-Programme auf den bisherigen Kleincomputern hat natürlich viel Arbeit gemacht. Deshalb ist es für die meisten Anwender des BIC wichtig, die bereits vorhandenen Programme nutzen zu können. Die prinzipielle Möglichkeit, auch am BIC mit handelsüblichen Magnetbandkassetten Programme lesen und speichern zu können (und auch handelsübliche Fernsehgeräte zur Ausgabe zu nutzen), hat deshalb viele begeistert. Allerdings ist es ein nicht geringer Aufwand, die Umstellung der BASIC-Diskette abzuarbeiten, besonders was die Bildschirmausgabe betrifft.

Natürlich gibt es auch zum Schriftsatz unterschiedliche Meinungen, die oft subjektiv gefärbt sind durch frühere Gewohnheiten. Der Wunsch, generell auch viel größere Buchstaben anzubieten, entspringt der Aufgabe des Einsatzes als Demonstrationsmittel im Unterricht. Hier sollten aber unserer Meinung nach die Schüler selbständig mit Unterprogrammen angepaßte Lösungen schaffen. Solche „Bibliotheken" könnten durchaus einen Bildungswert für sich haben.

Pseudografiksatz, Punktgrafik und hohe Rechengenauigkeit haben alle Anwender von der Astronomie-Simulation bis zur schnellen Darstellung von bewegten Bildern befriedigt.

Die Zuverlässigkeit unter Kabinettbedingungen bei ständigem Wechsel der Bediener wird als gut eingeschätzt. Das weite Feld der Einsatzmöglichkeiten als Steuergerät für das CAM-Training ist noch längst nicht ausgelotet und wartet auf weitere Bastler. Mit den nächsten Testberichten werden wir schlauer sein.

Dr. Joachim Adolphi
VEB Robotron-Meßelektronik
„Otto Schön" Dresden

Ein Tip für REDABAS-Nutzer

(rd 11/88, S. 31)

Zum oben genannten Artikel möchte ich Ihnen als REDABAS-Nutzer meine Meinung mitteilen. Die im angeführten Beispiel dargestellte Programmstruktur ist meiner Ansicht nach vom Aufbau her völlig unlogisch, läßt sich aber mit einer kleinen Änderung in die richtige Form bringen.

Wird das Beispiel verwendet, entsteht folgender Effekt:

1. Nach USE steht die Datei auf dem ersten Datensatz.
2. Die DO-WHILE-Schleife beginnt mit SKIP. Dadurch wird sofort auf den zweiten Datensatz gegangen. Der erste Datensatz ist in der Schleife nur mit GO-Befehl erreichbar.
3. Mit SAY-GET-READ-Konstruktionen werden die notwendigen Eingaben oder Ausgaben erreicht.
4. Es wird als letztes die Schleifenbedingung geprüft. Steht man auf dem letzten Datensatz, ist EOF nicht erfüllt. Die Schleife wird ein weiteres Mal durchlaufen, die Datei steht auf dem letzten Datensatz mit EOF gleich .T. Werden jetzt Daten eingegeben, wird EOF gleich .F.

Dieser Effekt gleicht der Eingabe von Daten mit APPEND.

Folgende einfache Lösung beseitigt das Problem sofort:

1. SKIP zu Beginn der Schleife streichen
2. SKIP neu nach den READ-Konstruktionen einfügen
3. Soll das Schleifeninnere mit einem beliebigen Datensatz beginnen, ist das mit GO einzustellen.

Die Verwendung von EOF unterscheidet sich damit bei REDABAS z. B. gegenüber der in PASCAL.

Oskar Schönherr
Rosmarienstr. 23
Karl-Marx-Stadt, 9030

Wissenschafts-Produktionsgemeinschaften

Die in vielen Kreisen der DDR gebildeten Wissenschafts-Produktionsgemeinschaften (WPG) haben das Ziel, durch die Zusammenführung von personellen und materiellen Kapazitäten von vorrangig Mittel- und Kleinbetrieben, deren Probleme bei der Rationalisierung innerbetrieblicher Prozesse lösen zu helfen, d. h. sie sind ein Mittel zur besseren Ausschöpfung der territorialen Ressourcen bei der Rationalisierung.

Im Kreis Dippoldiswalde entstand 1987 eine WPG, die sich vorrangig mit Problemen befaßt, die mit dem Einsatz der Mikroelektronik und Rechentechnik gelöst werden müssen. Die Mitgliedsbetriebe sind einige Betriebe eines ländlichen Gebirgskreises, der über ein verhältnismäßig geringes Potential zum Einsatz der Mikroelektronik und Rechentechnik vor allem in den kleineren Betrieben verfügt. Die Entwicklung einer zentralen Einrichtung auf diesem Gebiet ist hier notwendiges Erfordernis, da der Kreis außerhalb von industriellen Ballungsgebieten liegt. Das Aufgabenspektrum der gebildeten WPG umfaßt die Entwicklung von Anwendersoftware, die auf ganz konkrete betriebliche Erfordernisse zugeschnitten ist, die Qualifizierung von Werktätigen des Kreises zum Umgang mit Standardsoftware und in Zukunft die Lösung von Rationalisierungsaufgaben durch den Einsatz der Mikroelektronik in spezifischen Produktionsprozessen von kleineren Betrieben des Kreises, die über geringe oder sogar keine Kapazitäten zur Bewältigung derartiger Aufgaben verfügen. Die Arbeitsweise dieser WPG, die kein gleichgelagertes Vorbild in der DDR hat, entspricht einem kleineren Ingenieurbüro, für das die zu lösenden Aufgaben von einer Koordinierungsleitung festgelegt werden, wobei die Auswahl durch das Heranziehen ökonomischer Kriterien, d. h. das Verhältnis von Aufwand und Nutzen, erfolgt. Gemessen an den Aufgaben erscheint die Zahl von derzeit fünf in der WPG arbeitenden Ingenieuren und Wissenschaftlern als klein. Für die Entwicklung der WPG zu einem leistungsfähigen Instrument zur territorialen Rationalisierung durch den Einsatz einer modernen Schlüsseltechnologie ist nicht nur eine enge Zusammenarbeit der Mitgliedsbetriebe und personelle und materielle Unterstützung durch diese erforderlich, sondern auch ein enger Kontakt zum VEB Kombinat ROBOTRON, nicht nur um die materiellen Bedingungen im Stützpunkt der WPG zu verbessern. Der Stützpunkt der WPG befindet sich in der Ingenieurschule für Lebensmittelindustrie Dippoldiswalde, die gleichzeitig als Leitbetrieb der WPG fungiert. Über die WPG stehen damit die Kapazitäten der Ingenieurschule hinsichtlich dezentraler Rechentechnik auch anderen Betrieben des Kreises zur Verfügung. Andererseits arbeiten Mitarbeiter der WPG an der Lösung von Rationalisierungsaufgaben für die Ingenieurschule und wirken in Weiterbildungslehrgängen mit.

Dr. J. Zühlsdorf
Wissenschafts-Produktionsgemeinschaft Mikroelektronik-Rechentechnik Dippoldiswalde, Sitz: Ingenieurschule für Lebensmittelindustrie, Weißeritzstr. 11, Dippoldiswalde, 8230

Änderungsmitteilung

Wie uns mitgeteilt wurde, hat sich der Termin für die Fachtagung „Aufsetzmontage in der Leiterplattentechnik unter Berücksichtigung der Nacktchipverarbeitung" geändert. Sie findet mit neuem Termin vom 21. bis 22. Dezember 1989 statt (alter Termin: 27. bis 28. September 1989).

Anforderungen an die Implementierung von Software

Dr. Bernd Goes
Technische Universität Magdeburg
Sektion Informatik

Mit den Rechenanlagen der 3. Generation stehen technische Mittel zur Lösung anspruchsvoller Aufgaben aus vielen Bereichen des gesellschaftlichen Lebens, vor allem der Wirtschaft, der Wissenschaft und Technik zur Verfügung. Um diese Gerätetechnik voll zu nutzen, sind geeignete problemorientierte Systemunterlagen eine unabdingbare Voraussetzung. Diese Zielstellung erfordert nicht nur die Entwicklung, Erprobung und Bewertung von Software, sondern auch deren kontinuierliche Wartung im Sinne des Software-Lebenszyklus, um zu einer Intensivierung der Rechnernutzung zu gelangen. Während für Rechenanlagen der 1. und auch noch für die der 2. Generation nur relativ wenig allgemein nachnutzbare Software existierte, ist für EDVA der 3. Generation – insbesondere in Form von Programmpaketen, Programmsystemen und Programmbibliotheken – ein wesentlich breiteres Spektrum an problemorientierten Systemunterlagen verfügbar. Die hiermit gesammelten positiven Erfahrungen stimulieren die Nachnutzung dieser programmtechnischen Mittel. Gleichzeitig werden damit neue Bedürfnisse der Nutzer geweckt, und es wachsen die Aufgaben der ständigen Vervollkommnung dieser problemorientierten Software. Hierbei ist eine Konzentration auf Schwerpunkte der Anwendung erforderlich, die dem Profil der wissenschaftlich-technischen Arbeit der Forschungsbereiche bzw. -aufgaben von besonderer Aktualität entsprechen. Beim Bestimmen der Anforderungen an die Softwareentwicklung und -wartung ist von den Ergebnissen und Trends der internationalen Entwicklung der Informatik auszugehen.

Implementationsanforderungen

Für häufig genutzte Verfahren sind effektive Implementationen auf Computern mit jeweils hinreichenden Leistungsparametern anzustreben, die zudem die Lösung mit einer definierten Zuverlässigkeit liefern. Obwohl die Verfahren hinreichend exakt untersucht und beschrieben sind, ergibt sich bei ihrer rechentechnischen Realisierung dennoch eine Reihe von Schwierigkeiten. Um diese zu überwinden, besteht ein Weg darin, die jeweiligen Verfahren und Programme so zu projektieren und zu implementieren, daß sie problemorientiert sind, d. h. Problemspezifika berücksichtigen. Gleichzeitig werden auch in einem hohen Maße die Möglichkeiten und spezifischen Besonderheiten der Arithmetik des jeweiligen Rechnersystems und der maschinellen Realisierung der Algorithmen berücksichtigt.
Weiterhin erfordert die rechentechnische Projektierung und Implementation von Algorithmen in Gestalt von nachnutzbarer Software spezifische Arbeitstechniken und Verfahren der Datenverarbeitung, wie z. B. die Ausführung von Sortier-, Misch- und Suchprozessen, die Listenverarbeitung, die Arbeit mit Graphen, Adreßbüchern und Datenstrukturen, die Anwendung effektiver Zugriffsmethoden, die Nutzung unterschiedlicher Ebenen der Speicherhierarchie, die Fähigkeit des Umgangs mit beschränkten Ressourcen und nicht zuletzt die nutzerfreundliche Gestaltung von Mensch-Computer-Schnittstellen, um nur einige Aspekte zu nennen.

Anpassung des Computers an den Nutzer

Mit dem Übergang von einer Rechnergeneration zur nächsten wächst das Leistungsvermögen der Computer sowohl im Hinblick auf ihre Operationsgeschwindigkeit als auch die Kapazität ihrer Arbeits- und Peripheriespeicher, womit sich schritthaltend neue Einsatzbereiche eröffnen.
Parallel dazu vollzog sich ein Wandel in der Informatik in Form einer Neuorientierung. Während anfangs der Computer ausschließlich im Mittelpunkt aller Untersuchungen stand und vom Nutzer eine Anpassung erwartet wurde, bewirkt die Neuorientierung einen Wechsel der Prioritäten. Zunehmend rückt der Nutzer ins Blickfeld des Interesses und auch der Forschung, und verbunden damit wird eine Anpassung des Computers an den Nutzer gefordert. Dies setzt einerseits eine entsprechende Kenntnis bzw. Erforschung der Prozesse der Informationsverarbeitung im Menschen voraus und verlangt andererseits eine bewußte Gestaltung der Technologien geistiger Arbeit. Die Anpassung des Computers an seine Nutzer muß in mehrfacher Hinsicht erfolgen, um z. B.
– ergonomische Forderungen,
– Anforderungen rechnergestützter Technologien geistiger Arbeit und
– kognitiven Erkenntnissen
gerecht zu werden.
Dies setzt die Berücksichtigung der Einheit von Hard- und Software voraus, wobei die Nutzer-Hardware-Schnittstelle und die Nutzer-Software-Schnittstelle gleichermaßen und in ihrem Zusammenhang zu betrachten und zu gestalten sind. Beispiele, welche die Gestaltung der erstgenannten Schnittstelle betreffen, sind die
– Gestaltung und Belegung des Tastenfeldes für den alphanumerischen Dialog,
– Positioniereinrichtungen (Maus), Steuereinrichtungen,
– Fenstertechnik mit Überlagerungsfunktion und die
– Zoomfunktion zur Bildvergrößerung bzw. -verkleinerung.
Die zweitgenannte ist nicht nur mit der erstgenannten Schnittstelle verbunden, sondern vor allem auch mit der Architektur von Informationssystemen im allgemeinen und der Anwendungssoftware im besonderen. Die Software-Ergonomie als eine Brücke zwischen Informatik und Ergonomie bemüht sich um Voraussetzungen und Lösungen für die Projektierung informationsverarbeitender Systeme im allgemeinen und der Mensch-Softwareschnittstellen im besonderen. Softwareergonomische Fragestellungen entstehen immer dann, wenn die funktionellen Leistungsmöglichkeiten des Menschen und die Parameter der Nutzer-Software-Schnittstelle zu koordinieren sind. Kriterien sind dabei die Zuverlässigkeit, die Genauigkeit, die Arbeitserleichterung und anderes mehr. Beispiele, die die Gestaltung der zweitgenannten Schnittstelle betreffen, sind
– Menügestaltung für Mensch-Computer-Kommunikation,
– Gestaltung von Fehlermeldungen (in einer für den Nutzer „verständlichen" Sprache, d. h. auf seinem Kommunikationsniveau),
– Entwicklung und Verwendung von Modellen der Anwendungssoftware, die einen effektiven System- bzw. Softwareeinsatz ermöglichen,
– Einbeziehung der Kenntnisse, Fähigkeiten, Fertigkeiten, Eigenschaften und Verhaltensweisen der Nutzer in die Schnittstellengestaltung z. B. durch eine unterschiedliche Ausprägung der Menüs für verschiedene Nutzerklassen.

Nutzerklassen

Die Nutzer eines Systems bzw. Softwareproduktes werden aus der Sicht der Software-Ergonomie in Klassen eingeteilt, wobei die Vertreter der einzelnen Klassen unterschiedliche Anforderungen erfüllen und deshalb auch mit unterschiedlichen Modellen für Nutzer, unterschiedlichen Kommunikationsniveaus usw. bedient werden. Der einzelne Nutzer wird nicht starr und ein für allemal einer Klasse zugeordnet. Diese Einordnung nimmt der Nutzer selbst vor, und sie kann je nach Bedarf von ihm während des Nutzungsprozesses verändert werden.
Die Bildung von Nutzerklassen kann nach unterschiedlichen Gesichtspunkten erfolgen. Einer davon sind die Kenntnisse über die einzusetzenden informationellen Werkzeuge. In diesem Fall bietet sich z. B. folgende Klassifikation an:
1. Nutzer, die allgemeine Grundkenntnisse über die von der Software bereitgestellten Methoden besitzen
2. Nutzer, die eine Ausbildung für Ingenieure erfolgreich absolvierten
3. Nutzer, die eine Ausbildung für Ingenieure erfolgreich absolvierten und mit den Grundlagen der Computerarithmethik vertraut sind.
Neben den Informationskenntnissen könnten aber auch Kenntnisse des Nutzers über die Numerische Mathematik, über das Anwendungsgebiet und auch Erfahrungen mit dem Umgang mit Softwareprodukten zu einer weiteren Untersetzung der Nutzerklasse führen:
4. Nutzer, die zusätzlich mit der numerischen Verfahrenstechnik gut vertraut sind
5. Nutzer, die zusätzlich Spezialisten auf dem betrachteten Gebiet sind
6. Nutzer, die zusätzlich über umfangreiche Erfahrungen bei der Nutzung von Softwareprodukten verfügen.

Softwareprodukt zur Lösung linearer Gleichungssysteme

Unter Beachtung der obengenannten Forderungen

– effektive Implementation
– Zuverlässigkeit der Ergebnisse
– Nutzung geeigneter Techniken der Datenverarbeitung
– Berücksichtigung beschränkter Ressourcen
– Beachtung der Softwareergonomie
wurde vom Autor ein Softwareprodukt entwickelt, das die Computernutzung zur Lösung linearer Gleichungssysteme betrifft /4/.

Zur Lösung linearer algebraischer Gleichungssysteme existieren bereits verschiedene Programmpakete und -systeme /1–3/, /5–7/, die vorwiegend für allgemeine Aufgabenklassen anwendbar sind. In /4/ wird die Implementation von Algorithmen zur Lösung schlecht konditionierter und schwach besetzter linearer algebraischer Gleichungssysteme mit symmetrischer und positiv definiter Matrix unter Nutzung von Rechenanlagen des ESER behandelt, die die Spezifik der zu lösenden linearen algebraischen Gleichungssysteme und des Rechnersystems weitestgehend berücksichtigt. Es wurden ein Iterationsverfahren (konjugierte Gradienten mit Vorkonditionierung) und ein direktes Verfahren (Cholesky-Verfahren) sowohl als Operativ- als auch als Externspeichervariante implementiert. Des weiteren wird ein geeignetes Verfahren zur Minimierung der Hülle der Matrix zur Verfügung gestellt, mit dem eine wesentliche Erhöhung der Effektivität des Cholesky-Verfahrens erreicht werden kann. Für die oben beschriebenen Gleichungssysteme stellt das Cholesky-Verfahren das effektivste direkte Verfahren dar. Bei umfangreichen Systemen, die das Verwenden von Externspeichern erforderlich machen, hat sich insbesondere das Block-Cholesky-Verfahren bewährt, bei dem das Gleichungssystem in Submatrizen segmentiert wird. Da nur schwach besetzte Matrizen betrachtet werden, wurden die Lösungsalgorithmen so gestaltet, daß nur die Hülle bei der Realisierung der Verfahren berücksichtigt wurde. Unter Beachtung des Rechenzeitbedarfs für eine Cholesky-Zerlegung, der direkt proportional zum Quadrat der mittleren Bandbreite der Koeffizientenmatrix ist, wurde ein modifiziertes Cuthill-McKee-Verfahren angewendet, um mit der Minimierung der Hülle eine wesentliche Erhöhung der Effektivität des Cholesky-Verfahrens zu erreichen. Ein geeigneter effektiver Algorithmus zum Ermitteln eines Startknotens, der für den modifizierten Cuthill-McKee-Algorithmus benötigt wird, ist dem Minimierungsprogramm der Enveloppe vorangestellt. Die Permutation des Gleichungssystems $\underline{A}x=b$ in das Gleichungssystem $\overline{A}x=\overline{b}$ ist nur dann zu empfehlen, wenn mit der Permutation eine entsprechende Verkleinerung der Hülle der Koeffizientenmatrix erreicht wird. Falls a priori eine Abschätzung der Konditionszahl der Koeffizientenmatrix nicht möglich ist, kann diese mittels eines Algorithmus abgeschätzt werden. Die Konditionszahl kann für eine symmetrische, positiv definite Matrix als Quotient des größten und des kleinsten Eigenwertes berechnet werden. Der größte Eigenwert wird dabei aus einer Matrixnorm abgeschätzt. Die bereits vorhandene Zerlegung der Koeffizientenmatrix wird bei der Berechnung des kleinsten Eigenwertes nach der Methode der inversen Vektoriteration wirkungsvoll ausgenutzt. Lineare algebraische Gleichungssysteme mit schwach besetzter Matrix können auch vorteilhaft durch Iterationsverfahren gelöst werden, da nur die Matrixelemente, die ungleich null sind, in die Lösungsberechnung eingehen und durch eine kompakte Speicherung der Matrix schon bei einer Operativspeichervariante Gleichungssysteme hoher Ordnung gelöst werden können. Die Operativspeichervariante zeichnet sich dadurch aus, daß Zeilen der Koeffizientenmatrix, die mehrfach vorkommen, nur einmal gespeichert werden. Diese Maßnahme kann zu einer erheblichen Reduktion des Speicherbedarfs führen. Die Methode der konjugierten Gradienten ist besonders zur Lösung großer linearer algebraischer Gleichungssysteme geeignet, da sie keine speziellen Informationen über die Matrix des Systems erfordert wie z. B. die Kenntnis des minimalen und des maximalen Eigenwertes und außerdem über eine hohe Konvergenzgeschwindigkeit verfügt. Die Konvergenzgeschwindigkeit ist jedoch auch von der Konditionszahl der Matrix abhängig. Dieser Nachteil wurde mittels Vorkonditionierungsmaßnahmen überwunden. Ein Algorithmus der konjugierten Gradienten mit SSOR-Vorkonditionierung wurde sowohl als Operativspeichervariante als auch als Externspeichervariante implementiert. Die vom Autor implementierten Algorithmen zur Lösung schwach besetzter linearer algebraischer Gleichungssysteme mit symmetrischer und positiv definiter Systemmatrix berücksichtigen sowohl die speziellen Strukturen des Gleichungssystems als auch die Eigenschaften und Parameter der verfügbaren Gerätetechnik. Die Auswahl eines Verfahrens ist vom Nutzer in Abhängigkeit von den charakteristischen Größen der Systemmatrix wie Ordnung, Struktur, Kondition usw. und von den Parametern der verfügbaren Gerätetechnik wie verfügbare Hauptspeicherkapazität, Typ und Kapazität der Externspeicher, Operationsgeschwindigkeit usw. vorzunehmen. Ein automatischer Verfahrensauswahlprozeß, vor allem für die genannten Nutzerklassen **1.** bis **3.** und eine unterschiedliche Gestaltung des Kommunikationsniveaus, und anderes mehr, die aus der Sicht des Autors wünschenswert wären und bei zukünftiger Softwareentwicklung berücksichtigt werden sollte, sind in /4/ nicht enthalten.

Literatur

/1/ Autorenkollektiv: Nutzerhandbuch COSAR/84
Methodisch Diagnostisches Zentrum „Finite Elemente" TH Otto von Guericke Magdeburg, 1984
/2/ Autorenkollektiv: Programmbibliothek MABIF.AdW der DDR, Zentralinstitut für Isotopen- und Strahlenforschung, Leipzig 1981
/3/ Bunc, J. R.; Dongarra, J. J.; Moler, C. B.; Steward, G. W.: LINPACK-User's Guide. SIAM, Philadelphia 1989
/4/ Goes, B.: Zur Entwicklung von Software zur Lösung schlecht konditionierter und schwach besetzter linearer algebraischer Gleichungssysteme. Diss. A, TU Magdeburg 1987
/5/ Programmpaket NUMATH-2, H 4523-2004.
VEB Kombinat Robotron Dresden 1983
/6/ Programmpaket Numerische Mathematik, H 4513-0008.
VEB Kombinat Robotron Dresden 1979
/7/ Schaper, M.; Schieweck, N.: Programmsystem zur Lösung großer linearer Gleichungssysteme mit schwach besetzten Koeffizientenmatrizen. AdW der DDR, R-MECH-02/84
Berlin 1984

SOFTWARE-TECHNOLOGIE

Schnittstellengestaltung komplexer CAE-Software

Wolfram Lüneburg,
Günter Reinemann,
Wolfram Weiß
Technischen Hochschule
„Carl Schorlemmer"
Leuna-Merseburg,
Rechenzentrum

Rechnergestützte Problemlösungsprozesse werden gegenwärtig noch überwiegend von autarken Programmsystemen zur Lösung abgegrenzter Teilaufgaben realisiert. Diese „Insellösungen" ersetzen erfahrungsgemäß oft nur die herkömmlichen Arbeitsmittel bei organistorischen oder wissenschaftlichen Tätigkeiten. Bestehende Formen der Arbeitsteilung bleiben in der Regel unangetastet, so daß die Möglichkeiten der modernen Informationsverarbeitungstechnik nur zum Teil ausgeschöpft werden.

Stand und Anforderungen an komplexe CAE-Systeme

Die Notwendigkeit zur Schaffung integrationsfähiger Gesamtkonzepte für die Hard- und Softwareentwicklung ist deshalb ein dringendes Erfordernis /1/. Das schließt adäquate Aufgaben entsprechend der Leistungsfähigkeit der durch die Entwicklung der Mikroelektronik entstandenen Informations- und Kommunikationstechnologien ein und zwingt zur durchgängigen rechnergestützten Problembearbeitung.

Mit dem Integrationsgrad von Hard- und Software steigt die Komplexität der Informationsverarbeitungssysteme. Typische volkswirtschaftlich bedeutende Einsatzgebiete derartiger Systeme sind z. B.
– die flexible Fertigung und Steuerung der Produktion
– die durchgängige Gestaltung von Konstruktion, Technologie und Produktion
– die Automatisierung wissenschaftlicher Experimente und anderer Routineprozesse in der Forschung
– der Einsatz von Kommunikations-, Recherche- und Lernsystemen /2/.

Um in kurzer Zeit Lösungen und nachweisbare Effekte bei der durchgängigen rechnergestützten Gestaltung dieser Problemlösungsprozesse zu erreichen, müssen u. a. folgende Aufgaben gelöst werden:
– Schnittstellengestaltung spezialisierter Teilsysteme
– weitere Modularisierung der Hard- und Software
– Parallelverarbeitung in unterschiedlichen Teilsystemen
– Erhöhung der Wiederverwendbarkeit gefundener Lösungen /2/.
Softwaresysteme zur durchgängigen rechnergestützten Problembearbeitung sind durch die integrierte Informationsverarbeitung geometrischer, technischer, technologischer und ökonomischer Informationen gekennzeichnet /3/.
Nach /4/ werden für die Software
– eine einheitlich gestaltete Softwareentwurfskonzeption
– die Durchgängigkeit des Informationsflusses
– die Durchsetzung des Prinzips der Modularität
– die Strukturierung des komplexen Problemlösungsprozesses und
– einheitlich gestaltete Schnittstellen und Datenstrukturen
gefordert.
Neben den Anforderungen aus der Sicht des adäquaten und damit effektiven Einsatzes der Informationsverarbeitungstechnik sind natürlich auch
– Rechen- und Speichereffektivität
– das Vermeiden von Daten- und Programmredundanzen
– die Transparenz der Software
– die Ausbaufähigkeit der Software
– eine effektive Problembearbeitung durch den Nutzer
– die strenge Einteilung in hardwareabhängige und hardwareunabhängige Softwarekomponenten sowie
– eine fehlerfreie Funktion
zu gewährleisten.

Merkmale integrationsfähiger CAE-Software

Die Beherrschbarkeit komplexer Softwaresysteme erfordert für alle Lebensphasen eine geeignete vertikale und horizontale Strukturierung in Komponenten, welche bestimmte, möglichst standardisierte Funktionen ausführen. Die dabei entstehenden Relationen zwischen den Komponenten verschiedener Ebenen sind die Grundlagen späterer Schnittstellendefinitionen. Es existieren dazu verschiedene Konzepte. Diese werden von einem Systemverbund charakterisiert, der aus verscheidenen anwendungsspezifischen Bausteinen besteht, die über Standardschnittstellen integriert sind. Integrationsfördernde Merkmale sind
– ein modularer Systemaufbau der Hard- und Softwarekomponenten
– das Verwenden von Standardschnittstellen zum Datenaustausch
– die Nutzung von Datenbanksystemen bzw. die Arbeit mit flexiblen Datenstrukturen und -formaten sowie
– eine hohe Hardwareunabhängigkeit mittels Standardprogrammiersprachen (z. B. FORTRAN 77, PASCAL, C).
Generell kann eingeschätzt werden, daß die Beherrschung der Schnittstellen maßgeblich über die Realisierung integrierter Softwarekonzepte entscheidet. In der gegenwärtigen Entwicklungsetappe des Einsatzes solcher Lösungen ist dieser Faktor neben der Schaffung organisatorischer und technologischer Voraussetzungen für den Einsatz komplexer CAE-Systeme von ausschlaggebender Bedeutung. Integrationsfähige Systeme sind dabei zusätzlich durch eine ausgeprägte Gliederung in Modellalgorithmen und ihnen entsprechende rechnerinterne Darstellungen (RID) zur Realisierung verschiedener Objektmodelle gekennzeichnet.

Ausgewählte Schnittstellenstandards und ihre Einordnung

Die Definition und Standardisierung von Schnittstellen setzt die Modellierung des Problemlösungsprozesses und die entsprechende Abbildung in einem Informationsflußmodell voraus. Dazu liegen in den unterschiedlichen Zweigen und Strukturen der Volkswirtschaft z. T. sehr erheblich voneinander abweichende Bearbeitungsstände vor. Für den Bereich der stoffwandelnden Industrie ist in /5/ eine entsprechende Einschätzung gegeben. Eine analoge Einschätzung muß für die angebotenen CAD/CAM-Systeme gegeben werden. Sie unterscheiden sich in ihren Anwendungsbereichen und Fähigkeiten ebenfalls wesentlich voneinander /6/.
So wurden die vorhandenen Schnittstellennormen, die zum Teil aus sehr verschiedenen Gesichtspunkten entstanden, vorwiegend aus pragmatischen Gründen abgeleitet. Es muß festgestellt werden, daß aus der Sicht der Anwendungsklassen der stoffwandelnden Industrie der gegenwärtige Stand der Normung von Schnittstellen nicht ausreichend ist. Das hat unter anderem auch seine Ursache darin, daß z. B. im Maschinenbau schwerpunktmäßig andere Aufgaben (Konstruktion) als in der stoffwandelnden Industrie (Verfahrens- und Anlagenentwurf, Steuerung der Prozesse und Verfahren usw.) bearbeitet werden.
Die Zielstellungen der Schnittstellendefinitionen aller Ebenen sind dabei
– das Vermeiden von Informationsverlusten
– das Vermeiden von Mehrfacharbeit
– die Beherrschung komplexer technischer Anwendungen
– die Herbeiführung der Konsistenz der Datenbestände und
– die Reduzierung redundanter Datenhaltung /7/.
Die Hierarchie existierender Schnittstellennormen (Abb. 1) ist gekennzeichnet durch die
– Nutzerschnittstelle,
– Geräteschnittstelle,
– funktionelle Schnittstellen und
– Datenverwaltungsschnittstelle.
Für den Produktdatenaustausch existieren eine Reihe von Standards und Normungsvorschlägen (Tab.), die aber im wesentlichen nur die Geometriesicht auf ein Produkt berücksichtigen. Dabei nimmt IGES (Initial Graphics Exchange Specification) eine zentrale Rolle ein, deren Verwendung in Abbildung 2 dargestellt ist. IGES berücksichtigt neben den Geometriedaten auch Darstellungsdaten.
Ein Normungsvorschlag für den Austausch thermophysikalischer Stoffdaten stellt COSTAT (Codata Standard Thermodynamics) dar /8/.

Tab. Übersicht zu Produktdaten – Schnittstellennormen

IGES	**Initial Graphics Exchange Specification /6, 7, 13/** –CAD-Schnittstelle in Form von Pre- und Postprozessoren für Geometriedaten – schlechte Eignung für Flächen- und Körpermodelle – 1981 ANSI Y14, 26M
SET	**Standards d'Exchange et de Transfert /7, 14/** – französischer Normvorschlag, der im wesentlichen dem Funktionsumfang von IGES entspricht, aber im Format eine Effizienzsteigerung ergibt
VDAFS	**Verband der Automobilindustrie – Flächenschnittstelle /7, 15/** – speziell für die Übertragung von Freiformgeometrien – DIN 66301
VDAPS	**Verband der Automobilindustrie – Programmschnittstelle /17/** – systemneutrale Normteilgenerierungsprogramme in FORTRAN 77 – unterstützt Variantenkonstruktion in Abhängigkeit von Kenndaten – beschreibt 2D- und 3D-Objekte
STEP	**Standardization of Transfer and Exchange of Product Data /7, 16/** – in Entwicklung befindlicher Standard, an dem 17 Länder unter Führung der IGES-Vertreter der USA arbeiten
PDES	**Product Data Exchange Standard /16/** – in Entwicklung befindlicher umfassender Produktdatenstandard eines internationalen IGES-Komitees
CAD*I	**CAD Interfaces /12/** – im Rahmen des ESPRIT-Projektes laufende Entwicklung mit neuer Herangehensweise gegenüber IGES
COSTAT	**Codata Standard Thermodynamics /8/** – Normvorschlag der CODATA Task Group on Critically Evaluated Phase Equilibrium Data für den Austausch thermophysikalischer Daten von 1987 – berücksichtigt numerische Werte, den Quellennachweis, Meß- und Modellierungsbedingungen u. a. m.

Die Entwicklung von Standards für andere produktdefinierende Sichten, z. B. von Technologie (Material-, Fertigungsdaten) oder Organisation, befinden sich dagegen noch im konzeptionellen Stadium. Daraus resultiert oft einseitig die grafikorientierte Betrachtungsweise der durchgängigen rechnergestützten Problembearbeitung. Diese Strukturierung grafikorientierter Aufgabenklassen und deren Schnittstellen sind in Abbildung 3 dargestellt.

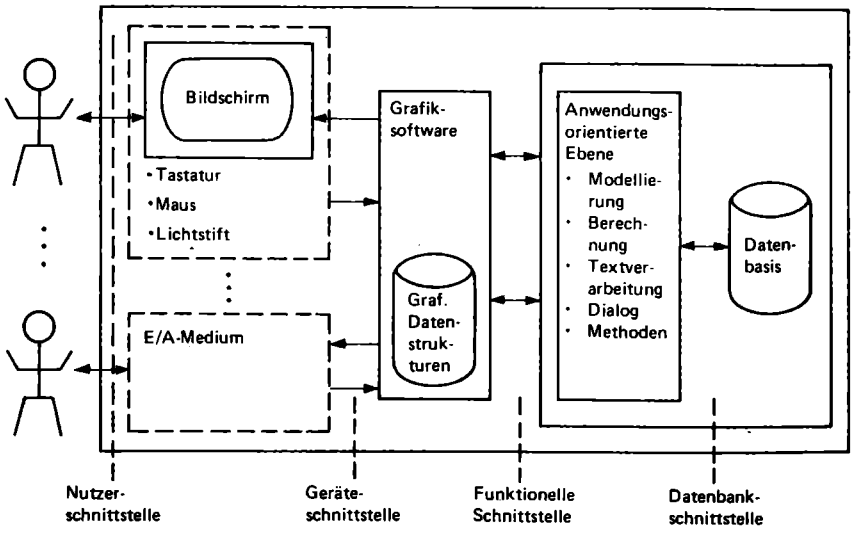
Um Daten zu verwalten, zu speichern und effektiv bereitzustellen, bedarf es eines Datenmodells. Datenmodelle sollen dabei weder von der abzubildenden Informationsstruktur noch vom rechnerinternen Speichermodell abhängen /9/. Entsprechend den wesentlichen Modelltypen (relationales, hierarchisches und Netzwerkmodell) existieren probleminvariante Datenverwaltungsschnittstellen zu Datenbanksystemen (DBS).

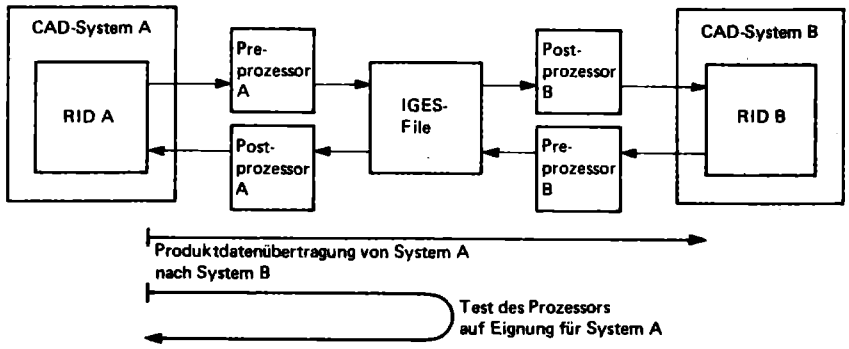
Eine Einschätzung verfügbarer CAD/CAM-Systeme zeigt, daß sie sich durch die verwendeten Datentypen, -modelle, -strukturen und -formate unterscheiden /7/.

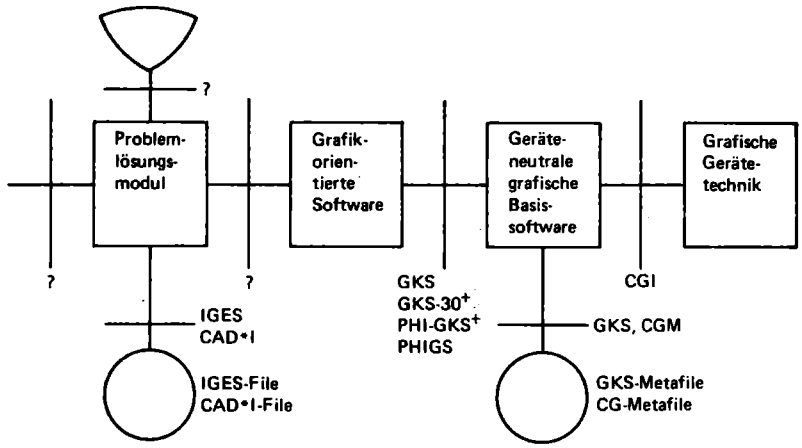
Beim Gestalten der hardwareunabhängigen Grafikschnittstelle hat sich gegenwärtig weitgehend das GKS-Konzet (Graphical Kernel System) in Folge seiner klaren, begrifflichen und funktionalen Definitionen durchgesetzt. Auf dieser Basis wurde das dazu kompatible GKS-3D entwickelt /10/.

Die Normenentwicklungen PHIGS (Programmers Hierarchical Interactive Graphic System) und PHI-GKS berücksichtigen dabei besonders die interaktive grafische Arbeitsweise. Zur Archivierung grafischer Ausgaben auf der Ebene der grafischen Programmierung existieren die Standards GKSM (Metafile) und CGM (Computer Graphics Metafile). Dabei werden bei GKSM Bildinhalt und Bildstruktur archiviert, dagegen bei CGM nur eine Momentaufnahme ohne Strukturinformationen /11/.

Der *grafische Assembler* CGI (Computer Graphics Interface) dient der Normung grafischer Befehlsgruppen für gewisse Geräteklassen, wobei auf Grund der zum Teil verschiedenen Geräteherstellerinteressen Kompatibilitätsprobleme existieren /11/.







? (noch) nicht standardisierte Schnittstelle
+ Entwicklungen die kompatibel zu GKS sind, sich aber durch differenzierten Leistungsumfang unterscheiden

Schlußfolgerungen

Die Schaffung komplexer CAE-Softwareprodukte erfordert den Einsatz von entsprechenden Schnittstellenstandards. Für Bereiche, die noch über keine Normen verfügen, ist ihre Entwicklung zu konzipieren, um Durchgängigkeit und Integration in der notwendigen Breite zu ermöglichen.

Mit der Verwendung von Standardschnittstellen lassen sich ingenieurmäßige Methoden in der Softwareentwicklung durchsetzen, die für sie selbst, aber auch über die Software, wichtige ökonomische Effekte auslösen. Als Beispiel sollen die Möglichkeiten zur Verschmelzung mehrerer Arbeitsaufgaben zu einer und die Verlängerung der Softwarelebenszeit genannt werden.

Die Entwicklung der bestehenden Normen basiert nicht auf einem einheitlichen Referenzmodell eines CAE-Systems und auch nicht auf einem einheitlichen Verständnis der angestrebten Ziele. Deshalb ist eine Harmonisierung des Systems von

Abb. 1 Basiskonfiguration grafikorientierter Systeme
Abb. 2 Produktdatenaustausch in CAD-Systemen
Abb. 3 Strukturierung grafikorientierter Aufgabenklassen und deren Schnittstellen

Schnittstellen dringend notwendig /11/. Da diese Aufgaben nicht nur unter logisch-inhaltlichen Gesichtspunkten zu betrachten sind, sondern auch stark von kommerziellen Interessen tangiert werden, sind kurzfristige Lösungen von der Industrie nicht zu erwarten. Um so mehr stellt sich dieses Aufgabengebiet als relevant für Forschungseinrichtungen der Akademie bzw. des Hochschulwesens heraus.
Wesentliche Effizienzsteigerungen beim Einsatz von Standardschnittstellen lassen sich mit der Übernahme dieser Softwarefunktionen durch die Hardware erreichen /11/. Dabei kann man mit weiteren Impulsen von der Entwicklung spezieller LSI- und VLSI-Schaltkreise und neuer Systemarchitekturen rechnen /11/.

Literatur
/1/ Gruhn, G.: Ingenieurwissenschaften, Mathematik und Informationsverarbeitung. Wiss. Berichte der TH Leipzig, Heft 7/1983, 10–19.
/2/ Niepel, H.: Datenaustausch zwischen Rechnermoduln. iir, 3(1987), 17–19.
/3/ Kehrer, B.; Grosch, H.-S.: edv-aspekte, 4(1985)2, 15–19.
/4/ Klose, J.; Kochan, D.: edv-aspekte, 4(1985)2, 2–4.
/5/ Weiß, W.; Reinemann, G.: Rechnergestütztes Ingenieurwesen – Stand und Perspektiven in der stoffwandelnden Industrie. Chem. Technik 37(1985)7, 273–278.
/6/ Richter, R.; Kreischer, U.: Die Standardisierung des Produktdatenaustausches. Arbeitsbericht DS/MA/1/87 des IIR der AdW der DDR, Berlin, 1987.
/7/ Schuster, R.; Vöge, E.: VDI-Berichte 570.1, 1985, 53–74.
/8/ Wilhoit, R. C.; Maczynski, A.: COSTAT Rules for Transmitting Thermodynamic Data Phase Equilibrium Version. Systembeschreibung, 1987.
/9/ Rauchhaus, K.-H.; Reinemann, G.; Weiß, W.: CAD-Datenbanken – Besonderheiten, Stand und Perspektiven ihrer Nutzung für die rechnergestützte Ingenieurarbeit in der stoffwandelnden Industrie. Chem. Technik 38(1986)6, 229–234.
/10/ Noll, S.; Poller, J.; Rix, J.: Harmonisierung grafischer Standards. PHI-GKS sichert die Kompatibilität von PHIGS zu GKS-3D. Informatik-Spektrum 10(1987)5, 239–245.
/11/ Encarnacao, J. L.: Forschungsthemen und Trends in der Grafischen Datenverarbeitung. XI. Internationaler Kongreß über Anwendungen der Mathematik in den Ingenieurwissenschaften, Berichte 2, 36–45, Weimar, 1987.
/12/ Schlechtendahl, E. G.: Specification of a CAD I Neutral File for Solids, Version 2.1. Springer, Berlin, 1986, Research Reports ESPRIT, Project 322, CAD Interfaces (CAD I), Vol. 1.
/13/ Initial Graphics Exchange Specification. Proposed American National Standard, Product Definition Standard, Version 3.0, National Bureau of Standards, Washington, 1986.
/14/ Standard d'Exchange et des Transfert, Specification Rev. 1.1, Aerospatiale, 1984.
/15/ VDA-Flächenschnittstelle (VDAFS), Version 1.0. Verband der Deutschen Automobilhersteller, BRD, 1983.
/16/ Vatterott, G.: Schemawandler – Ein Weg zum standardisierten Produktdatenaustausch, iir-report. Nr. 8/1987.
/17/ Weissflog, U.: Produktionsdatenaustausch, IGES und seine Alternativen. VDI-Berichte 570.5, Düsseldorf (1985), 147–160.

SOFTWARE-TECHNOLOGIE

Effektive Softwareentwicklung mittels Trennung von Programm- und Datenstruktur

Frank Jauch
Forschungszentrum für Bodenfruchtbarkeit Müncheberg der Akademie der Landwirtschaftswissenschaften der DDR, Bereich Jena

Bei der Softwareentwicklung ist die Forderung, die Aufgabe umfassend und genau zu formulieren, ein zentrales Problem. Die hierzu erforderliche tiefe Problemanalyse kommt in der Regel zu kurz. Auch zwingen fehlende Programmierkapazitäten und der Termindruck oft zu einem übereilten Beginn der eigentlichen Programmierarbeiten. Die rasante Entwicklung des wissenschaftlich-technischen Fortschritts einerseits und der relativ hohe Zeitbedarf bei der Entwicklung von Software andererseits führen oft dazu, daß das gerade fertiggestellte Softwareprodukt veraltet und aktualisierungswürdig ist. So wird für die ständige Pflege und Weiterentwicklung des entstehenden/„fertigen" Softwareprodukts in der Regel zu wenig Kapazität geplant.
Bei der Konzeption und Algorithmierung von Software sollte somit ein Aspekt im Vordergrund stehen:
Änderungen und Erweiterungen müssen leicht eingearbeitet werden können – verbesserte Rechentechnik und neue fachliche Erkenntnisse sollten nicht zwangsläufig zur völligen Neuprogrammierung eines Projektes führen.

Konzeption des PC-Projektors ISBO

Trennung von Programm und Datenstruktur

Bei der Entwicklung des Softwarepaketes ISBO (Informationssystem Bodenführung) des Forschungszentrums für Bodenfruchtbarkeit der ADL der DDR wurde diesem Aspekt zentrale Bedeutung beigemessen. Kernstück stellt dabei das früher bereits vorgestellte Datenspeicherkonzept UDS /1/ dar. Grundgedanke war, die für die Bodenführung in der Landwirtschaft benötigten großen Datenmengen inhaltlich und in ihrer Struktur weitgehend aus den eigentlichen Programmen herauszunehmen und außerhalb in Daten- und Parameterdateien abzulegen (Abb. 1). Für diese Dateien wurde schon mit dem Konzept ein umfassender Änderungsdienst vereinbart, der natürlich weitgehend ohne Einfluß auf die eigentlichen Programme bleiben sollte. Erweiterungen oder Änderungen finden nicht gleich in neuen Programmversionen, sondern erst in neuen Dateiversionen ihren Ausdruck. So wurden während der Überleitung des Paketes mehrere Versionsänderungen von Dateien vorgenommen, ohne dabei die eigentlichen Programme in ihrer Funktion zu beeinträchtigen.
Natürlich stellt dieses Konzept an die Programmierung erhöhte Anforderungen, da die Definition und der Bezug zu den meisten benötigten Daten nicht im Programm festgeschrieben werden können, sondern dynamisch über sachlogische Kurzworte aus einer externen Strukturdatei zugewiesen werden (Abb. 2). Der erhöhte Aufwand war jedoch gerechtfertigt, wie die Erfahrungen während der Erprobung und Überleitung zeigten.
Die Laufzeit der Programme erhöhte sich mit diesem Konzept nicht wesentlich. Lediglich nach dem Start des Programms erfolgt durch die dynamische Strukturzuweisung der Variablen eine leichte Verzögerung.
Die im UDS verwendeten Datentypen sind allgemeingültig, d. h., es wurden keine Typdefinitionen verwendet, die sprachspezifischen Charakter haben. Damit ist die Nutzung in verschiedenen Programmiersprachen leicht möglich.

Vielfältiger Zugriff zu den Daten

Die freie und universelle Nutzung der entstehenden Datenfonds war eine zweite zentrale Forderung im Projekt. Neben den im Projekt festgeschriebenen Datenzugriffen, die zu Standardlisten bzw. Standardtabellen führen, wurde ein freier Zugriff zu den Dateien geschaffen. Damit hat der Nutzer vielfältige Möglichkeiten, eigene Tabellen oder Formulare zu entwerfen und die Auswertungen seinen spezifischen Bedingungen anzupassen.
Für die Nutzung der Daten in Standardsoftwarepaketen wurde die Konvertierung realisiert. Die Belieferung weiterer Datenformate (Kalkulationsprogramm) wird perspektivisch auch möglich sein. Für die 16-Bit-PC wird dieser Zugriff auch für grafische Darstellungen (Trends) entwikkelt.

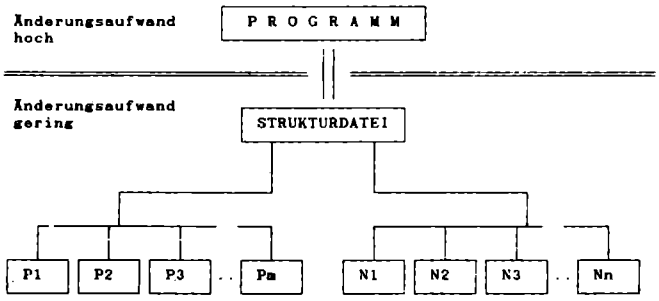
Programmiersprachen

Da die strukturelle und inhaltliche Definition der Daten des Projekts weitgehend extern erfolgt und die verwendeten Datentypen nicht an eine spezielle Sprache gekoppelt sind, war es

VERANSTALTUNGEN

VII. Stochastik-Kolloquium: Modell- und rechnergestützte Produktionsplanung und -steuerung

Termin: 6.–8. 9. 1989
Ort: Rostock
Veranstalter:
Wilhelm-Pieck-Universität Rostock, Sektion Sozialistische Betriebswirtschaft
Schwerpunkte:
– Produktionsvorbereitung und -durchführung (insbesondere in Maschinenbaubetrieben)
– Instandhaltung von Produktionsanlagen
– Transport-, Umschlag- und Lagerprozesse
Organisatorische Hinweise:
Interessenten wenden sich bitte an folgende Adresse: Wilhelm-Pieck-Universität Rostock, Sektion Sozialistische Betriebswirtschaft, Prof. Dr. W. Runge, Klement-Gottwald-Str. 6, Rostock 1, 2500



P1...Pm: Parameterdateien
N1...Nn: Nutzerdateien

Aufbau der wahlfreien Dateien P1...Pm und N1...Nn

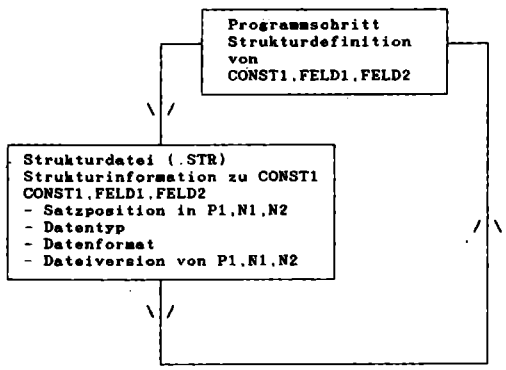
Anzahl Sätze	Datum	Version	Satzlänge	Frei	...		Kennsatz
KENNBYTE	FELD1	FELD2	FELD3			FELDj	Satz 1
KENNBYTE	FELD1	FELD2	FELD3			FELDj	Satz 2
KENNBYTE	FELD1	FELD2	FELD3			FELDj	Satz k

Programmfunktion

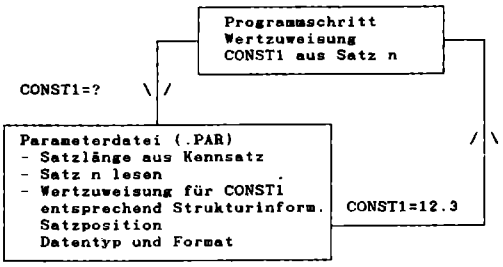
"Berechnung der Größe ERG nach ERG = CONST1 * FELD1 + FELD2"

ERG : gesuchte Größe
CONST1: num. Parameter aus der Parameterdatei P1
FELD1 : num. Größe aus der Nutzerdatei N1
FELD2 : num. Größe aus der Nutzerdatei N2

1. Schritt
- dynamische Strukturdefinition für CONST1,FELD1,FELD2



2. Schritt - Wertzuweisung für CONST1,FELD1,FELD2



Analog FELD1 und FELD2 behandeln

3. Schritt - ERG berechnen

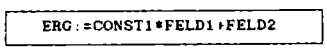


Abb. 1 Grundkonzept für die Softwareentwicklung von Projekten mit mittleren und großen Datenmengen

Abb. 2 Datenzugriff unter UDS (Beispiel)

nicht notwendig, für das Projekt eine bestimmte Programmiersprache festzulegen. Es wurde in T-PASCAL, BASIC und FORTRAN programmiert. Der Zugriff zum UDS erfolgt über einheitliche sprachspezifische Zugriffsroutinen.

Die Wahl des Datenspeicherkonzepts für Projekte mit mittlerem und hohem Datenerfassungsaufwand sollte folgende Kriterien berücksichtigen:

– Der Datenspeicher sollte flexibel sein – offen für Änderungen und Erweiterungen
– die Nutzung der Daten in Standardsoftwarepaketen sollte möglich sein
– sprach- oder compilerspezifische Konzepte unterliegen einem nicht beeinflußbaren Änderungsdienst der Hersteller
– ähnliches gilt für Datenbanken, bei denen die Aufwärtskompatibilität nur vom Hersteller selbst gesichert ist.

Die Berücksichtigung dieser Aspekte im UDS ist Voraussetzung dafür, die in ISBO entstehenden Datenfonds sowohl auf 8-Bit-PC als auch auf 16-Bit-PC (mit qualitativ besseren Programmen) zu nutzen. Der in den Pflanzenproduktionsbetrieben entstehende umfangreiche Datenfonds des ISBO hat somit Aussicht, langfristig nutzbar und somit nützlich zu sein.

Literatur

/1/ Jauch, F.; Gernat, H.: UDS-Nutzerfreundliches Datenbanksystem für PC.
rechentechnik/datenverarbeitung 4/88

SOFTWARE-TECHNOLOGIE

EXPRESS-Entwicklungs- und Steuersystem für REDABAS-Anwendungen auf 16-Bit-PC

Dr. Eberhard Börner
VEB Mansfeld-Generallieferant Metallurgie
– Erzprojekt Leipzig –
im Mansfeld Kombinat „Wilhelm Pieck"

Für 16-Bit-PC (EC 1834, A 7150 oder Kompatible) wurde ein Entwicklungs- und Steuersystem für große Anwenderprogrammsysteme unter REDABAS-4 (bzw. kompatiblen DB-Betriebssystemen) geschaffen. Damit wird eine wirkungsvolle Entwurfs- und Programmierunterstützung gegeben, die die bisherigen Möglichkeiten wesentlich erweitert, indem

– eine direkte Abbildung des funktionellen Entwurfs des gesamten Anwendungssystems in eine spezielle Steuerdatei mit nahezu beliebig vielen Funktionen und Hierarchieebenen erfolgen kann und damit
– eine große Einheitlichkeit, Übersichtlichkeit und Flexibilität bezüglich Menübildgestaltung und Ablaufsteuerung gesichert wird,
– eine leichte Handhabbarkeit bei der Entwicklung und Anwendung gegeben ist (Testbedingungen, Nutzerführung),
– günstige Voraussetzungen für Weiterentwicklung, Pflege und Wartung geschaffen werden und
– eine Reduzierung der Entwicklungszeit des Gesamtsystems um etwa 30 Prozent erreicht wird.

Die REDABAS-Programmierung beschränkt sich auf das Erstellen von Prozeduren in Prozedurdateien, die auf allen Ebenen des Menübaumes aktivierbar sind. Die Prozeduren erfüllen funktionell abgegrenzte Verarbeitungsfunktionen. Maximal drei Parameter können übergeben werden. In den Prozeduren sind Fremdprogramme aufrufbar (RUN). Es entstand ein praxisnahes Werkzeug zur Datenbank-Softwareentwicklung, das in allen Phasen des Lebenszyklus einer REDABAS-Anwendung vorteilhaft einsetzbar ist und damit eine Lücke in der effektiven Anwendung des meistgenutzten DBBS für 16-Bit-PC schließt.

Das System EXPRESS liegt gegenwärtig in der Version 1.1 vor.

Systementwurf

Entsprechend dem Phasenmodell des Software-Lebenszyklus wird bereits bei der Analyse der Anforderungen an die Anwenderlösung, u. a. mit

– Beschreibung des Problems aus Anwendersicht
– Analyse der Funktionen
– Formulieren der Wünsche des Auftraggebers in Systemfunktionen und Leistungsparametern
– Problemspezifikation durch den Systementwickler und der Definition der Anwenderlösung, u. a. mit
– Festlegung der externen Benutzerschnittstellen
– Ausarbeitung von Modellösungen
– Durchführbarkeitsuntersuchungen
– Beurteilung von alternativen Lösungsmöglichkeiten mit Auswahl einer geeigneten Variante
– Spezifikation der Systemanforderungen und
– Planung des Gesamtprojektes, eine hierarchische Strukturierung des Gesamtsystems vorbereitet. Eine immer tiefer gehende Spezifikation des Funktions- und Leistungsumfanges der Subsysteme führt zu einem Top-Down-Entwurf des Gesamtsystems und dessen Schnittstellen. Sobald greifbare Ergebnisse vorliegen, sollten diese im EXPRESS-Programmiersystem festgehalten werden, damit erste Tests des logisch-funktionellen Entwurfs unter Einbeziehung der künftigen Nutzer durchgeführt und ausgewertet werden können (Rapid Prototyping). Änderungen bei der schrittweisen Verfeinerung des Entwurfs bereiten keinerlei Schwierigkeiten und können erneut untersucht werden. Besonders hervorzuheben ist, daß der hierarchische Entwurf auch die Schnittstelle zum künftigen Anwender darstellt, d. h. der Systementwurf bestimmt die Bildschirmsteuerung. Durch die Auswahl eines Menüpunktes kann in eine Unterebene mit maximal zehn Auswahlmöglichkeiten verzweigt werden. Dies geschieht solange, bis keine Untergliederung mehr sinnvoll ist. An diesen Stellen werden beim Abarbeiten kleinere, gut abgegrenzte Verarbeitungsprozeduren ausgeführt, die sinnvoll zusammengefaßt in Prozedurdateien anzulegen sind. In jeder Hierarchieebene sind demzufolge je nach Problemlage Auswahl- und Prozedurknoten möglich.

Parallel zur Strukturierung des Gesamtsystems werden die erforderlichen Datendateien aufgebaut und schrittweise die Anwenderprozeduren geschrieben und getestet. Mit dieser Vorgehensweise sind folgende Zielstellungen realisierbar:

1. Zerlegung in logisch-zusammenhängende, überschaubare und klar definierte Aufgaben

2. Schrittweise Vervollkommnung des Systementwurfs ist in allen Entwicklungsphasen unkompliziert möglich

3. Veränderungen der Ablaufsteuerung führen zu keinen Prozedurveränderungen, soweit diese nicht interne Abläufe in den Prozeduren selbst betreffen

4. Erzeugen einer projektbegleitenden Kurzdokumentation mittels Ausgabe der Systemstruktur sowie der einzelnen, kommentierten Prozedurdateien.

Abbildung des Systementwurfs in die Steuerdatei MASTER.DBF

Abbildung 1 zeigt die allgemeine Projektstruktur, von der auszugehen ist. Die Struktur enthält Auswahl- und Prozedurknoten, wobei die Strukturierung an jeder Stelle mit der Festlegung eines Prozedurknotens beendet werden kann, und zwar genau dann, wenn eine weitere Untergliederung nicht sinnvoll ist, d. h., wenn die eigentliche Prozedurabarbeitung an dieser Stelle beginnen soll (z. B.: Eingabe, Aktualisierung, spezifische Berechnungen, Löschen, Sortieren, Ausgeben einer bestimmten Liste ...). Abfrage- bzw. Auswahlknoten haben stets eine Untergliederung entweder in weitere Abfrageknoten oder in Prozedurknoten oder in beide.

Es ist zweckmäßig, für die Hauptfunktionen (teil1, teil2, ...)

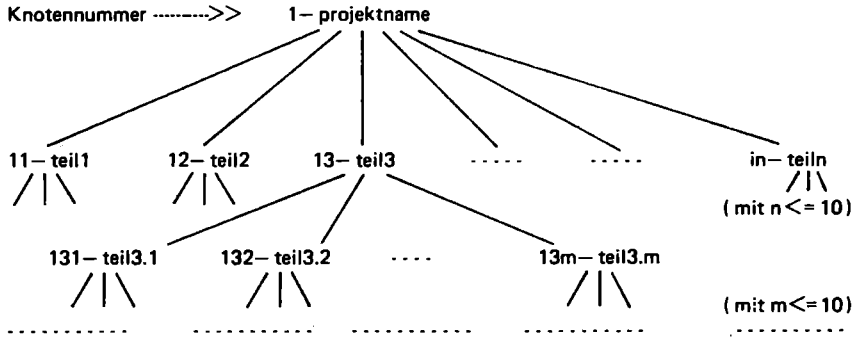


Abb. 1 Allgemeine Projektstruktur

EXPRESS
EINGABE MENÜBILDAUFBAU UND PROGRAMMAUFRUF MIT PARAMETERÜBERGABE

O^X ----------->>> KNOTEN-NR. 1^X

BEZ1 HAUPTFUNKTION bzw. HAUPTPROJEKTNAME^X

BEZ2 projektspezifikation^X

PRDAT PDATEI^X

PRNAME MENU^X PRNAME2 BASIS

PAR1
PAR2
PAR3

EDIT ||<A:>||MASTER ||Satz: 1/5 || ||

Abb. 2 Eingabebildschirm für Auswahlknoten Teil 1 – Hauptmenü (angekreuzte Angaben wurden eingegeben)

MENÜBILDAUFBAU

FUNKTION 1^X	11^X
FUNKTION 2^X	12^X
FUNKTION 3^X	13^X
FUNKTION 4^X	14^X
FUNKTION 5^X	15^X
FUNKTION 6^X	16^X
FUNKTION 7^X	17^X
FUNKTION 8^X	18^X
DOKUMENTATIONSERSTELLUNG DES GESAMTSYSTEMS IM ÜBERBLICK^X	190^X
ALLGEMEINE DATENBANKFUNKTIONEN^X	100^X

EDIT ||<A:>||MASTER ||Satz: 1/5 || ||

Abb. 3 Eingabebildschirm für Auswahlknoten Teil 2 – Hauptmenü (angekreuzte Angaben wurden eingegeben)

jeweils eigene Prozedurdateien, die bis zu 32 Prozeduren aufnehmen können, vorzusehen, um eine übersichtliche inhaltliche Gliederung zu erreichen. Somit ist es möglich, die Anzahl der offenen Dateien, die keine Prozedurdateien sind, zu erhöhen und demzufolge komplexere Probleme zu bearbeiten.

Die Anzahl der Knotenpunkte ist praktisch unendlich (9 999 999 999); aber die Anzahl der Ebenen ist bei Einhaltung der Dezimalklassifikation mit 9 begrenzt, was für praktische Aufgabenstellungen völlig ausreichend ist. Auf jeder Ebene sind die Funktionen jeweils mit 2×60 Stellen beschreibbar.

Im einzelnen ist in die Steuerdatei MASTER.DBF folgendes einzugeben:

– Knotennummer maximal 10stellig (der Hauptknoten hat stets die Nr. 1)

– Bezeichnung 1 und 2 der jeweiligen Funktion mit maximal 2×60 Stellen (bei Auswahlknoten wird die Bezeichnung auf dem Bildschirm ausgegeben; ansonsten dient diese nur der Kurzdokumentation der Prozedur)

– Prozedurname (bei Auswahlknoten ist dies MENU; bei Prozedurknoten jeweiliger Prozedurdatei einzutragen, in der die Prozedur steht – siehe auch oben)

– Prozedurname (bei Auswahlknoten ist dies MENU; bei Prozedurknoten jeweiliger Prozedurname mit eventueller Parameterübergabe)

– 2. Prozedurname (wahlweise kann eine 2. Prozedur an diesem Knotenpunkt aktiviert werden. Bei Auswahlknoten wird diese Prozedur nach der Entscheidung für einen Menüpunkt ausgeführt, wobei ein 4stelliger Parameter (vgl. unten) übergeben wird, der der jeweiligen Entscheidung zugeordnet ist. Somit ist das EXPRESS-Programmiersystem auch für große hierarchische Abfrage- und Recherchesysteme einsetzbar. Bei Prozedurknoten wird die Prozedur nach der 1. Prozedur ausgeführt, was z. B. für abschließende Arbeiten (z. B.: Neuindexierung nach Erfassung, spezielle Auswertungen) sinnvoll nutzbar ist)

– Parameterübergabe an 1. Prozedur (durch 3 10stellige Parameter ist eine gewisse Universalität der Prozedurnutzung erreichbar. Die Parameterübergabe ist nicht an die Prozedur MENU möglich)

– Vorgängerknoten (bei Knotennr. 1 ist dies 0)

– Auswahltext 1, ..., 10 mit jeweiliger Nachfolgeknotennummer (der Auswahltext entfällt bei Prozedurknoten. Sinnvoll ist es, die Dezimalklassifikation einzu-

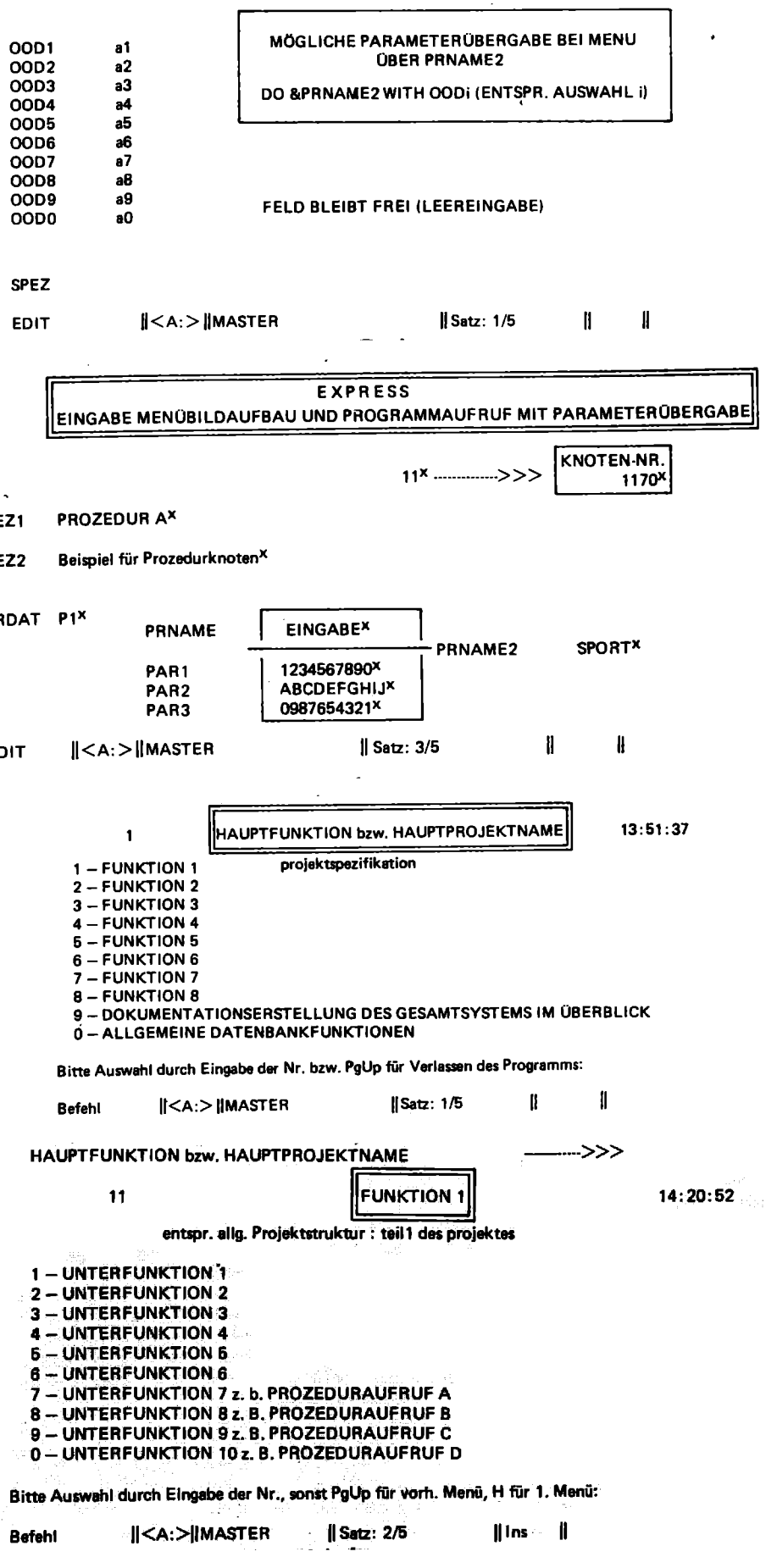


Abb. 4 Eingabebildschirm für Auswahlknoten Teil 3

Abb. 5 Eingabebildschirm für Prozedurknoten Teil 2 (Teil 2 und 3 bleiben leer)

Abb. 6 Beispiel für Hauptmenübild (von EXPRESS erzeugt)

Abb. 7 Beispiel für erstes Menübild nach Auswahl von Funktion 1

halten. Deutet der Nachfolgeknoten auf eine Prozedur, so wird empfohlen, durch Anhängen einer zusätzlichen Null dies kenntlich zu machen, soweit die maximale Stellenanzahl noch nicht überschritten ist – z. B.: 170. Jeder Knoten des obigen Diagramms ist als ein Satz der Steuerdatei mit allen Vor- und Rückverweisen auszuweisen. Eine automatische Kontrolle ist noch nicht realisiert und soll einer kommenden Version vorbehalten bleiben. Inkonsistenz der Steuerdatei ist jedoch im Test sehr schnell herauszufinden und zu beseitigen)

– Übergabecode 1, ..., 10 (die Übergabe ist – wie oben beschrieben – nur an die 2. Prozedur, die nach MENU abgearbeitet werden kann, möglich. Der Code kann auch leer sein, jedoch ein Parameter – dann ohne Bedeutung – ist in der 2. Prozedur notwendig. Die Anwendung ist auch zu empfehlen zur Zuordnung der Datenbankdateien zu den einzelnen Arbeitsbereichen mit Eröffnung der Datenbank- und Indexdateien. Diese Zuordnung würde dann einheitlich für alle Prozeduren gelten).

Durch eine übersichtliche Eingabemarke für die Datei MASTER.DBF wird der Erfassungsprozeß unterstützt (Abb. 2 bis 5).

Anwendung

Parallel zur inhaltlich-funktionellen Strukturierung des gesamten Anwendungssystems sind die Datenstruktur festzulegen, zu erproben sowie die entsprechend festgelegten Prozeduren zu erstellen und zu testen. In die REDABAS-Prozeduren können mit RUN ... EXE-Files eingebunden werden, die in einer anderen Programmiersprache geschrieben worden sind. Somit kann das EXPRESS-System auch die Basis für kompliziertere Anwendungen bilden. Mit DO EX wird das Hauptprogramm aufgerufen, das die Steuerung des gesamten Anwendungssystems übernimmt. EX sollte sinnvollerweise in den Namen des Hauptprojektes umbenannt werden. Die Steuerdatei wird im Arbeitsbereich 10 eröffnet. Dieser Datei werden die erforderlichen Angaben wie Prozedurdateiname, Prozedurname, Texte, Nachfolgeknotennummer usw. entnommen, der Menüdurchlauf sowie die Aktivierung der richtigen Prozeduren organisiert. Der Nutzer erhält jeweils ein einheitlich aufgebautes Bildschirmbild angezeigt (Abb. 6 und 7), aus dem er mittels Eingabe einer Ziffer 1, ..., 9 bzw. 0 10 Auswahlmöglichkeiten hat. In allen Prozeduren entfallen SAY...- und GET...-Befehle und die daraus im allgemeinen resultierenden DO CASE...-Befehle, soweit diese nicht zu internen Steuerzwecken gebraucht werden.

Bei Prozeduraufrufen erfolgt eine Rückversicherung durch das Steuerprogramm, ob diese Prozedur richtig ausgewählt worden ist (z. B. bei Löschfunktionen u. ä. bedeutsamen, die Datei(en) verändernden Funktionen). Jederzeit, außer während der Prozedurausführung, kann mit PgUp das übergeordnete Menü erreicht werden, mit H jederzeit das Hauptmenü, das dem Datensatz von MASTER.DBF mit Knotennr. 1 entnommen wird. Eine Spezialfunktion erlaubt sogar die Veränderung der Hauptdatei und die anschließende Weiterarbeit mit veränderter Steuerung, ohne das Steuersystem EXPRESS zu verlassen. Verlassen wird EXPRESS vom Hauptmenü durch die Eingabe von PgUp.

Prozeduränderungen können beim Auftreten eines Fehlers bei speicherresidentem SIDEKICK sofort in der Prozedur vorgenommen werden, ebenfalls ohne Verlassen des EXPRESS-Systems.

Einschätzung

In allen Phasen des Software-Lebenszyklus ist das vorliegende Programmiersystem erprobt worden und führte stets zu einer Erhöhung

– der Übersichtlichkeit des Gesamtsystems
– der Akzeptanz durch den Nutzer
– der Änderungsfreundlichkeit und
– der Abarbeitungssicherheit.

Erste Ergebnisse der Projektierung können sofort ausgetestet werden. In einer weiteren Ausbaustufe ist vorgesehen, Prozeduraufrufe bei Bedarf nur dann zuzulassen, wenn ein entsprechender Code eingegeben worden ist. Ebenso werden dann Funktionsaufrufe unterer Hierarchieebenen, die häufig verwendet werden, über einen vom Nutzer festzulegenden Code aufrufbar sein.

Vergleichbare Systeme, z. B. analoge Komponenten zu dem international verbreiteten System dBASE III PLUS, haben nicht annähernd diese Leistungsfähigkeit und Universalität in der Anwendung. Es wird eingeschätzt, daß der dort verfügbare Programmgenerator für den professionellen Einsatz nur bedingt geeignet ist. Nur eine Hierarchieebene ist verfügbar, Parameterübergaben sind nicht möglich.

Insgesamt kann eingeschätzt werden, daß das vorliegende Entwicklungswerkzeug ein neues Hilfsmittel zu einer weiteren Effizienzsteigerung in der Projektierung großer Datenbank-Anwendersysteme unter Nutzung moderner Software-Entwicklungsmethoden darstellt. Der Einarbeitungsaufwand zur Beherrschung des EXPRESS-Systems kann als gering eingeschätzt werden. Die Abarbeitungszeit bei der Nutzung dieses Metasystems steigt gegenüber der herkömmlichen Organisation der REDABAS-Programmierung nicht wesentlich an.

Interessenten wenden sich bitte an VEB Mansfeld-Generallieferant Metallurgie, Erzprojekt Leipzig, Abt. JETE, Brandvorwerkstr. 72, Leipzig, 7030

SOFTWARE-TECHNOLOGIE

MGEN – ein Maskengenerator

Bernd Blacha
VEB
Datenverarbeitungszentrum
Karl-Marx-Stadt

Mit dem erweiterten Einsatz von Kleinrechentechnik wird Werkzeugen, die den Dialog mit dem Nutzer unterstützen, eine große Bedeutung zukommen. Zunehmend werden Nutzer ohne tiefere EDV-Kenntnisse mit der neuen Technik konfrontiert. Solche Nutzer erwarten für die von ihnen zu lösenden Aufgaben speziell angepaßte Nutzerumgebungen. Diese Umgebungen führen den Nutzer, entlasten ihn von routinemäßigen Arbeiten und geben, wenn nötig, Hilfestellung.

Mit dem vorgestellten Maskengenerator wird Softwareentwicklern die Möglichkeit geboten, solche Nutzerumgebungen aufzubauen. Der Maskengenerator wurde für das Betriebssystem MUTOS1630 implementiert. Portierungen liegen für die Betriebssysteme WEGA, MUTOS1700/MUTOS1834 und DCP vor.

Mit Hilfe von Masken kann eine komplexe Schnittstelle zwischen dem Bildschirm (Erfassen der Daten) und dem Anwenderprogramm (Verarbeiten der Daten) gestaltet werden. Eine Maske zerlegt den Bildschirm in einzelne, mit Eigenschaften versehene Felder. Nach den Eigenschaften steuert ein Modul den Zugriff auf die Felder. Er entscheidet, ob der Cursor in bestimmte Bereiche des Bildschirms gelangen kann, nimmt die Nutzerdaten entgegen und unterstützt die sofortige Prüfung der Daten. Das Anwenderprogramm erhält die Daten, indem es auf die Namen der Eingabefelder bezug nimmt. So ist die physische Steuerung des Eingabevorgangs von der logischen Verarbeitung der Daten getrennt.

Das vorgestellte Werkzeug unterstützt das Erstellen (Maskendefinierer) und Verarbeiten (Maskenprozessor) von Bildschirmmasken. Der Maskendefinierer ist ein interaktives Kommando. Als Maskenprozessor steht ein Satz von Programmfunktionen zur Verfügung. Zur Zeit ist die sprachliche Schnittstelle die Programmiersprache C. Versionen zur Nutzung der Sprachen FORTRAN sowie MODULA-2 sind in Arbeit.

Zum Leistungsumfang des Maskendefinierers gehören

– das interaktive Erstellen von Bildschirmmasken,
– das Abspeichern dieser Masken im Filesystem,
– das Warten erfaßter Masken

VERANSTALTUNGEN

3. IFAC-Workshop Echtzeitprogrammierung

Termin: 18. bis 20. 10. 1989
Ort: Berlin
Schwerpunkte:
– Software engineering in real-time programming
– Specifications of real-time software systems
– Languages for real-time programming
– Programming methodologies and environments for real-time systems
– Parallel computers in real-time application
– Expert systems for real-time applications

Organisatorische Hinweise: Anfragen sind zu richten an die KDT, WGMA, Clara-Zetkin-Str. 115/117, PSF 1315, Berlin, 1086

– die Dokumentation erfaßter Masken.

Zum Leistungsumfang des Maskenprozessors gehören

– das Anzeigen der Masken,

– die Cursorsteuerung auf dem Bildschirm,

– die Ein- und Ausgabe von Daten zwischen Bildschirm und Anwenderprogramm,

– eine feld- und maskenorientierte Reaktion auf Nutzereingaben,

– das Unterstützen von Maskenüberlagerungen und

– die Organisation von Tabellenbereichen zum Erfassen und Verarbeiten tabellarischer Daten.

Maskenarten

Eine Maske wird von folgenden Parametern charakterisiert:

– Größe auf dem Bildschirm (Anzahl der Zeilen, Anzahl der Spalten)

– Art, Position, Größe und Inhalt der Maskenfelder in dieser Maske

– Steuerinformationen für das Verarbeiten der Maske.

Jedes Maskenfeld erhält beim Erstellen der Maske ein Feldattribut. Dieses Attribut beschreibt die Eigenschaften des Feldes. Typische Feldeigenschaften sind:

– Wählbare Videoeigenschaft (erhöhte Feldhelligkeit, inverse, blinkende oder unterstrichene Darstellung des Feldes)

– Feld ist ungeschützt

– Feld besitzt Eingabezwang.

Beim Verarbeiten der Maske unterscheidet der Maskengenerator zwischen geschützten und ungeschützten Feldern. Er realisiert die Cursorsteuerung zu letzteren und erlaubt die Eingabe von Daten in diese Felder. Die geschützten Felder können vom Nutzer nicht erreicht und somit auch nicht modifiziert werden.

Die Maske in Abbildung 1 hat eine Größe von 16 Zeilen und 40 Spalten. Felder der Maske sind u. a.:

a) „Arbeitszettel"

b) „--------------------"

c) „`````````````"

Alle Felder der Maske, die das Zeichen ` enthalten, sind ungeschützt. Die anderen Felder der Maske besitzen die Feldeigenschaft geschützt. Als eine spezielle Maskenart unterstützt der Maskengenerator das Erstellen und Verarbeiten von Tabellenmasken. Abbildung 2 zeigt eine solche Maske.

Tabellenmasken bestehen aus einem normalen Maskenteil (geschützte und ungeschützte Felder) und können ein bis fünf Tabellenbereiche enthalten. Ein solcher Bereich besteht aus mindestens zwei Zeilen. Im Bild ist

```
                M A S K E N P A R A M E T E R

Name der Maske               :  az0

Beendigungstasten            :  F1 F2 F3 F4 F10

Maskengroesse
Anzahl der Maskenzeilen      :  16
Anzahl der Maskenspalten     :  40
Anzahl der Maskenfelder      :  21

Anzahl der Tabellen          :  keine


Feldbeschreibung
Feldnr    Feldname   Zeile   Spalte   Laenge   Eigenschaften

  1                    2        4       13     prot high
  2                    2       26        7     prot
  3       jahr         2       36        2     prot
  4                    3        4       35     prot
  5                    5        4        9     prot
  6       name         5       14       12     unprot input
  7                    6        4        9     prot
  8       vorname      6       14       12     unprot input
  9                    7        4        9     prot
 10       monat        7       14        2     unprot input

                                :
                                :
                                :

Bild der Maske az0


-
-    Arbeitszettel          Jahr  :   88
-    -----------------------------------
-
+    Name    : ````````````
-    Vorname : ````````````
-    Monat   : ``              [ 1 - 12 ]
-    Dekade  : `               [ 1 -  3 ]
-
1    ----------------------------------
-
-    F1  - ENDE
-    F2  - Arbeitszettel erfassen
-    F3  - Arbeitszettel drucken
+    F10 - HILFE
-
 ----+----1----+----2----+----3----+----4
```

```
Arbeitszettel                  -                      Erfassen
--------------------------------------------------------------

Name: Mueller               Vorname: Heinz            Jahr: 88

KK  : 777      Werk : 02    Bereichsnr. : 10000       Monat: 03

/----|-----------|-------------|---LAT---|--------|-+-|----|--\
| Tag | Person.nr | Auftrags.nr |    AdA  |    h   | - | VA | 4 |
----------------------------------------------------------------
  11  |   11111   |  00001101   | 000115  | 00875  | ` | `  | `
  12  |   11111   |  00001101   | 000115  | 00875  | ` | `  | `
  13  |   11111   |  ````````   | 000941  | 00875  | ` | `  | `
  _`  |   `````   |  ````````   | ``````  | 00875  | ` | `  | `



F1: ENDE  F2: Einfuegen F3: Loeschen F8: Vor F9: Zurueck F10: Posit
```

Abb. 1 Ausschnitt aus der Liste einer Maske

Abb. 2 Maske mit Tabellenbereich während der Tabellenarbeit

die Arbeit in der Tabelle dargestellt.

Maskendefinition

Nach dem Aufruf des Kommandos madef erscheint auf dem Bildschirm ein Grundmenü (Abb. 3). Um eine neue Maske zu erzeugen, wählt man in diesem Menü die Funktion a. Der Maskendefinierer erfragt nun notwendige Maskenparameter. Ein Maskenparameter ist z. B. die Größe der Maske. Um die Angaben zu erhalten, bietet der Definierer dem Anwender weitere Menüs an. Der erste Schritt beim Erstellen einer neuen Maske ist das Erfassen des Maskengerüstes. Dazu wird der Bildschirm in der Größe der Maske mit Füllzeichen ausgegeben. In diesem sogenannten Skelett erfaßt der Anwender die Maske so, wie sie später auf dem Bildschirm erscheinen soll (Abb. 4).

Der Maskendefinierer interpretiert eine Zeichenkette zwischen zwei Füllzeichen als Maskenfeld. Es erhält das Feldattribut geschützt. Die Skeletterfassung ist mit Funktionstasten unterstützt. Mit diesen können zum Beispiel Tabellen eingerichtet oder wieder entfernt werden.

Der Skeletterfassung schließt sich im allgemeinen die Definition der Feldeigenschaften an (Funktion d im Grundmenü des Maskendefinierers). Geht der Anwender zur Eigenschaftsvergabe über, erscheint die bearbeitete Maske auf dem Bildschirm. Mit der Cursorbewegung gelangt er auf jedes Feld der Maske. Im unteren Teil des Bildschirms werden die Eigenschaften des jeweiligen Feldes angezeigt. Mit Funktionstasten kann der Anwender Eigenschaften für das Feld vergeben oder zurücknehmen.

Aus folgenden Feldeigenschaften kann sich das Feldattribut eines Maskenfeldes zusammensetzen: UNPROT, CLICK, NODISP, INPUT, HIGHL, INVERS, UNTER, BLINK. Die Eigenschaften HIGHL, INVERS, UNTER, BLINK werden als Videoeigenschaften bezeichnet. Sie sind von der konkreten Hardware abhängig, und somit für die einzelnen Maskengeneratoren unterschiedlich realisiert. Abbildung 5 zeigt die möglichen Feldeigenschaften eines Maskenfeldes im Überblick.

Die Eigenschaft UNPROT erfordert für das Feld einen Feldnamen. Über diesen Feldnamen kann bei der Maskenverarbeitung auf die Daten aus diesem Feld zugegriffen werden. NODISP unterdrückt das Anzeigen der Nutzerdaten im Feld. INPUT erzwingt eine Eingabe in das Maskenfeld. Für geschützte Felder ist die Vergabe der Eigenschaft CLICK möglich. Beim Verarbeiten der Maske gelangt der Nutzer mit dem Cursor auf diese Felder, kann deren Inhalt aber nicht modifizieren. Die für ein Blickfeld vergebene Videoeigenschaft ist nur dann wirksam, wenn sich der Cursor auf diesem Feld befindet.

Tabellenfelder besitzen keine Videoeigenschaft. Ebenso sind die Eigenschaften CLICK und NODISP für diese Felder unzulässig.

Mit den Funktionen s und r speichert der Anwender die bearbeitete Maske in seiner Datenbasis. Der Maskengenerator unterstützt den Einfluß der Maske in das Anwenderprogramm. Mit der Funktion h wird die Datenstruktur der Maske in ein Quelltextstück umgesetzt.

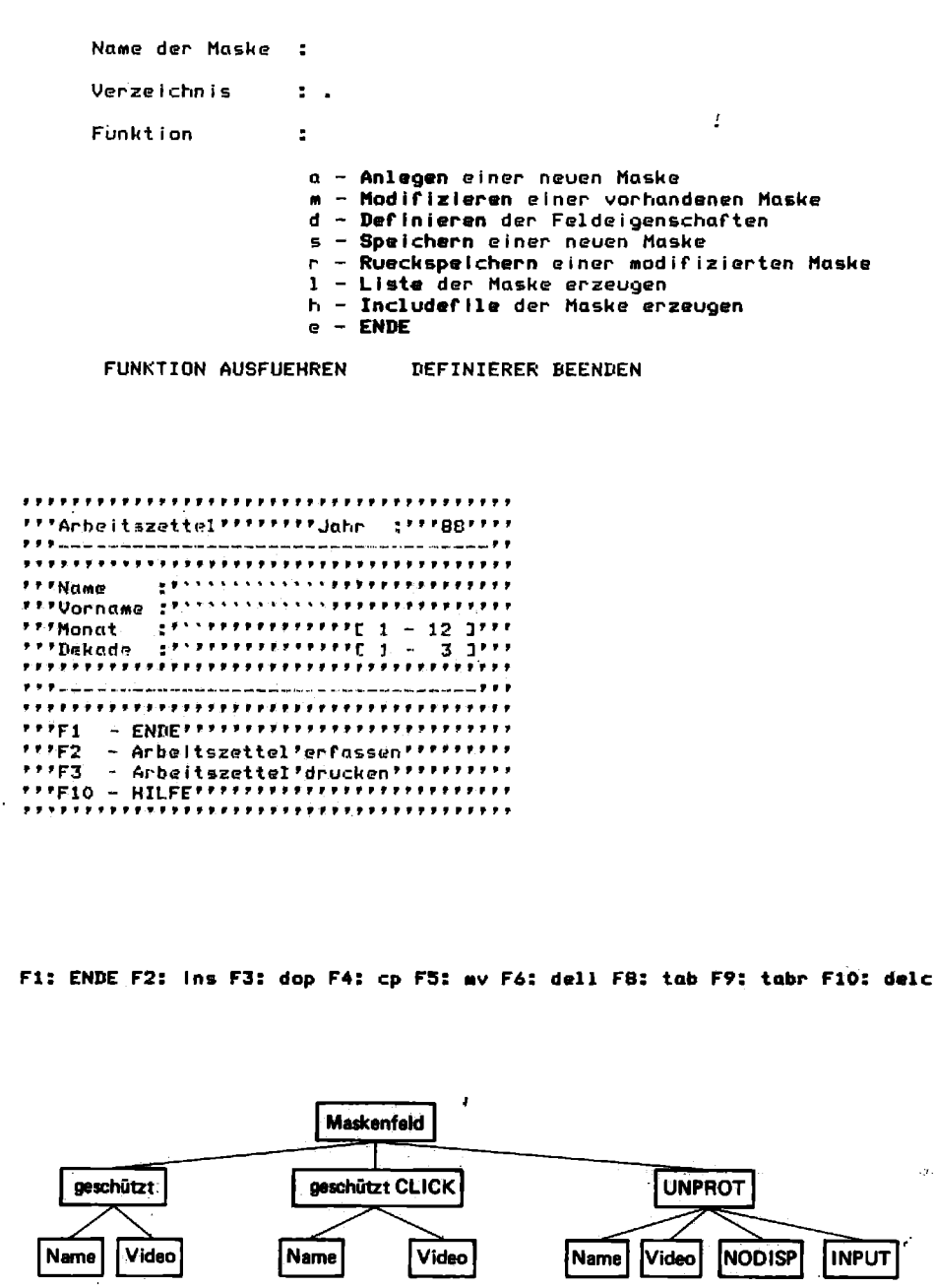


Abb. 3 Grundmenü des Maskendefinierers
Abb. 4 Skeletterfassung
Abb. 5 Überblick zu den Feldeigenschaften

Maskenverarbeitung

Um die erstellten Masken zu verarbeiten, muß ein Anwenderprogramm erstellt werden. Die Funktionen, die die Maskenverarbeitung möglich machen, stehen in Bibliotheken zur Verfügung. Im DCP sind das zum Beispiel die Bibliotheken mgen.lib und mgent.lib.

Das Verarbeiten einer Maske beginnt mit der Funktion mpinit. Mit dieser Funktion wird die Maske für die Ausgabe auf den Bildschirm vorbereitet. Entsprechend ihrer Größe ist sie frei auf dem Bildschirm positionierbar. Der Programmierer kann Felder der Maske modifizieren (mpputc), und ihnen Programmfunktionen zuordnen (mpusr). Diese Funktionen heißen Nutzerfunktionen. Außerdem kann er für die Maske eine Nutzerendefunktion definieren. Sind die notwendigen Vorarbeiten abgeschlossen, übergibt das Anwenderprogramm die Steuerung an den Maskenprozessor (mpgo). Die initialisierte Maske erscheint auf dem Bildschirm.

Der Maskenprozessor realisiert

die Cursorsteuerung zu den ungeschützten Feldern und den Clickfeldern der Maske. Wird ein Feld verlassen, welches eine Nutzerfunktion besitzt, so aktiviert der Maskenprozessor diese Funktion. Dadurch kann unmittelbar während der Arbeit mit der Maske der Feldinhalt der einzelnen Maskenfelder geprüft werden (feldorientierte Interaktion). In der Nutzerfunktion kann der Programmierer den größten Teil der Prozessorfunktionen anwenden. Er kann den Maskeninhalt verändern (mpcreaf, mprmf, mpdelf), die Eigenschaften einzelner Felder modifizieren (mpchatt), die aktive Maske mit einer anderen Maske überlagern (mpload Abb. 6) und anderes mehr. Durch ihren Rückkehrcode teilt die Funktion mit, ob das Feld verlassen werden darf oder nicht.

Wird während der Verarbeitung der Maske ein Feld in einem Tabellenbereich erreicht, aktiviert der Maskenprozessor automatisch den Tabellenprozessor. Voraussetzung dafür ist, daß das Anwenderprogramm mit der Bibliothek mgent.lib verbunden wurde.

Beendet der Nutzer die Arbeit mit der Maske, wird in die Nutzerendefunktion verzweigt (maskenorientierte Interaktion) und der Maskenprozessor prüft die Felder mit der Feldeigenschaft INPUT auf einen nichtleeren Eintrag. Danach geht die Steuerung an das Anwenderprogramm zurück. Die Maske ist für den Maskenprozessor aber noch aktiv. Das Anwendprogramm kann mit mpgo wieder zur Maskenarbeit übergehen. Die Funktion mpend beendet die Arbeit mit der aktiven Maske.

Zum Leistungsumfang des Maskengenerators gehören eine einfache Maskenüberlagerungstechnik und die Unterstützung der Arbeit mit Tabellen.

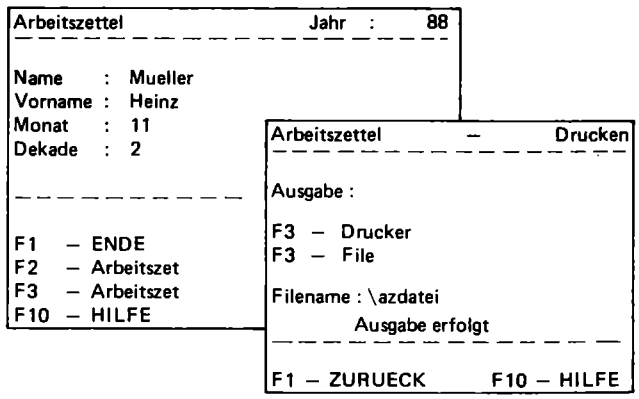
Maskenüberlagerung

Die Funktion mpload lädt eine Maske und initialisiert für sie einen neuen Maskenprozeß. Dazu werden die Prozeßvariablen der aktiven Maske ausgelagert. Zu diesen gehört auch der Teil des Bildschirminhalts, der von der neuen Maske benutzt wird. Erscheint die Maske auf dem Bildschirm, werden alle Videoeigenschaften der überlagerten Maske zurückgesetzt. Dem neuen Maskenprozeß stehen alle Funktionen des Maskenprozessors zur Verfügung. Der Programmierer kann ihn z. B. ebenfalls wieder überlagern. Wird die Arbeit der Maske beendet, lagert der Maskenprozessor die alte Maske automatisch wieder ein. Deshalb muß die Verarbeitung der Nutzerdaten aus der überlagernden Maske in deren Nutzerfunktion oder deren Nutzerendefunktion erfolgen.

Die Zahl der Überlagerungen ist nicht begrenzt. Alle Überlagerungen werden nacheinander wieder abgetragen.

Tabellenverarbeitung

Während der Arbeit mit einer Tabelle, stehen alle ungeschützten Felder der Tabellenzeile für Eingaben zur Verfügung. Die Cursorsteuerung erstreckt sich bis zum Beenden der Tabellenarbeit (Funktionstaste F1) nur auf die aktive Tabelle. Wird die aktuelle Tabellenzeile verlassen, geht der Cursor in die nächste Zeile des Tabellenbereiches. Diese ist ebenso strukturiert wie die vorangegangene und wird neue Tabelleneingabezeile. War es die letzte Zeile in der Tabelle,




```
mpinit      -   Initialisieren einer Maske ( Maske ist im Filesystem )
mpini       -   Initialisieren einer Maske ( Maske ist im Programm )
mpgo        -   Uebergabe der Steuerung an den Maskenprozessor
mpend       -   Beenden der Maskenarbeit
mpload      -   Ueberlagern einer aktiven Maske
mploa       -   Ueberlagern einer aktiven Maske
mpusr       -   Zuweisen der Nutzerfunktionen und der Nutzerendefunktion
mtabusr     -   Zuweisen der Nutzerfunktionen und der Zeilenendefunktion
mpexit      -   Beenden der Maskenarbeit ohne Abschlusspruefungen
mpex        -   Beenden der Maskenarbeit mit Abschlusspruefungen
mtabexit    -   Ende der Tabellenarbeit ist Ende der Maskenverarbeitung
mpgetc      -   Verarbeiten der Zeichen in einem Feld
mpgetf      -   Verarbeiten des benannten Feldes
mpputc      -   Ausgabe von Zeichen in ein Feld
mpputoc     -   Ausgabe von Zeichen in ein Feld mit einem Offset
mpcreaf     -   Erzeugen eines neuen Maskenfeldes
mprmf       -   Feld aus der Maske entfernen
mpdelf      -   Feldinhalt loeschen
mpchatt     -   Feldeigenschaft veraendern
mpset       -   Cursor auf ein bestimmtes Feld setzten
mpnoinp     -   Keine Feldeingabezwangpruefung beim Beenden der Maske
mpdiag      -   Einschalten der Maskenprozessordiagnose
mpcd        -   Maskenverzeichnis festlegen
```

```
cmafu( name , laenge, flag ,taste)
char *name;                         /* Feldname */
int laenge;                         /* Feldlaenge */
int flag;                           /* Feldeigenschaft */
int taste;                          /* betaetigte Taste */
{
int rc=0;                           /* Rueckkehrcode der Nutzerfunktion */

if( taste == CR ) {

        if( strcmp("cma0",name) == 0 )  mpexit( 1 );
        if( strcmp("cma1",name) == 0 )  mpexit( 2 );
        if( strcmp("cma2",name) == 0 )  mpexit( 3 );
        if( strcmp("cma3",name) == 0 )  mpexit( 4 );

}

return rc;
}
```

Abb. 6 Darstellung einer Überlagerung
Abb. 7 Übersicht zu den Funktionen des Maskenprozessors
Abb. 8 Funktionsweise des Maskenprozessors
Abb. 9 Darstellung einer Nutzerfunktion

```
mpinit("maske1",5,20); /* Initialisieren der Maske */
   "
/* Platz für Vorarbeiten:
   – Modifikation von Feldinhalten (mpputc, mpputoc)
   – Anschluß der Nutzerfunktionen und der Nutzerendefunktion an die Maske (mpusr)
*/
   "
rc=mpgo("feld1"); /* Steuerung geht an Maskenprozessor */
   "
/* nach Maskenprozeß:
   – verarbeiten der eingegebenen Daten (mpgetf, mpgetc)
   – bei Fehlern mpgo
*/
   "
mpend(0); /* Beenden des Maskenprozesses */
   "
   "
```

blättert der Tabellenprozessor eine Seite weiter. Dadurch steht die gesamte Tabelle wieder für die Datenerfassung zur Verfügung. Die Tabellenverarbeitung wird von Funktionstasten unterstützt. Diese realisieren zum Beispiel das seiten- oder zeilenweise Positionieren innerhalb der Tabellendaten, sowie das Löschen und Einfügen von Zeilen in den Datenbestand.

Die Schnittstelle zwischen dem Anwenderprogramm und den Daten, die in einer Tabelle erfaßt wurden, ist eine Datei. Den Namen für diese Datei vergibt der Maskendefinierer beim Einrichten der Tabelle. Findet der Tabellenprozessor diese Datei, so wird sie eröffnet und die darin enthaltenen Daten werden in der Tabelle angezeigt. Jedem ungeschützten Feld der Tabelle kann mit der Funktion mtabusr eine Nutzerfunktion zugeordnet werden. Der Tabellenprozessor bedient diese analog zu den schon beschriebenen Nutzerfunktionen. In Abbildung 7 sind alle Funktionen zusammengestellt, um mit dem Maskenprozessor zu arbeiten. Abbildung 8 erläutert noch einmal die prinzipielle Funktionsweise des Maskenprozessors.

Mit dem Anschluß von Nutzerfunktionen an Felder einer Maske, kann das Anwenderprogramm sofort auf Nutzereingaben reagieren.

Abbildung 9 zeigt eine Nutzerfunktion. Sie ist allen Clickfeldern einer Maske zugeordnet. Die Funktion enthält als Aufrufargumente den Namen, die Länge und die Eigenschaften des Feldes, sowie die Taste, die betätigt wurde. Sie kontrolliert, ob der Nutzer die CR-Taste betätigt hat. In diesem Fall wird das Verarbeiten der Maske mit einem dem Feld zugeordneten Rückkehrcode beendet.

Systeminformation

Für die eingangs erwähnten Betriebssysteme wird mit den einzelnen Maskengeneratoren das gleiche Leistungsspektrum verwirklicht. Der Maskendefinierer ist ein ausführbares Programm. Sein Hauptspeicherbedarf beträgt etwa 50 KByte.

Die Funktionen des Maskenprozessors werden in Bibliotheken ausgeliefert. Diese Bibliotheken sind einem bestimmten Arbeitsspektrum entsprechend konfiguriert. Die Funktionen des Prozessors haben einen Hauptspeicherbedarf von 15 KByte (ohne Tabellenverarbeitung) und 23 KByte (mit Tabellenverarbeitung). Um diese Größen erweitert sich das Anwenderprogramm, wenn mit dem Maskenprozessor gearbeitet wird. Zusätzlich zu diesen zwei Bestandteilen ist eine Bibliothek mit Bildschirmbasisfunktionen Bestandteil der Auslieferung. Mit ihnen kann auf dem Bildschirm positioniert und gelöscht werden. Außerdem realisieren die Funktionen einen direkten Zugriff auf den Bildschirm. Als Dokumentation werden neben der Systeminstallationsschrift eine Anwenderbeschreibung und eine Nutzeranleitung geliefert. Alle Teile der Dokumentation sind auf magnetischen Datenträgern vorhanden.

SOFTWARE-TECHNOLOGIE

UDIS – Universelles Dialogsystem

Rudolf Klar
VEB
Datenverarbeitungszentrum
Karl-Marx-Stadt

Dialogprogramme ermöglichen eine anwenderfreundliche Gestaltung des Nutzerinterfaces von Programmsystemen und Programmen mit intensivem Datenaustausch. Die Entwicklung leistungsfähiger Dialogprogramme ist aufwendig. Für die Definition von Bildschirmmasken steht in der Regel ein Maskengenerator zur Verfügung. Bei hinreichender Abstraktion läßt sich auch die Dialogsteuerung, die Erstellung der Bildschirmroutinen und der Datenaustausch verallgemeinern. UDIS stellt für den Dialogentwurf, die Dialoggenerierung und die Dialogverwaltung universell einsetzbare Programme und Funktionen zur Verfügung. UDIS ist für MUTOS 1630 (K1630) und DCP (EC 1834, A7150) fertiggestellt. Eine Überführung auf die Systeme

- WEGA (P8000)
- MUTOS 1700 (A7150)
- MUTOS 1834 (EC 1834)
- MUTOS 1800 (K1840)

ist geplant.

Das Programmsystem ist in C geschrieben, Anschlüsse für MODULA-2 und FORTRAN sind vorgesehen.

Konzept

Dialogprogramme unterscheiden sich bezüglich der Programmsteuerung von herkömmlichen Programmen. Die umfangreiche Steuerung einer Vielzahl von Bild- und Verarbeitungsroutinen erfolgt direkt oder indirekt anhand der im Dialog angeforderten Daten. Die Steuerung der Aufruffolge ist in herkömmlichen Programmen Bestandteil des Verarbeitungsalgorithmus (Alternativen), im Programmtext verstreut angeordnet und hängt nur von wenigen Daten ab (Aufrufparameter, Verarbeitungskennzeichen, Satzart u. ä.). Für Programme, deren Routinen überwiegend in Abhängigkeit von Steuerdaten aufgerufen werden, ist die externe Steuerung günstiger. Die Steuerungsübergabe zwischen den

Abb. 1 Dialogbild während der Generierung des Szenariums (es wurde ein Helptext angefordert)
Abb. 2 Dialogbild während der Pflege des Szenariums (links die Aktualisierungs- und rechts die Servicefunktionen)

```
S Z E N A R I U M anlegen          ? fordert Hilfe an
  Name: szenar1                    * aktiviert das Service

Folgesteuerung einer Szene     Der Typ der Szene bestimmt
Name der Szene: sz3            die Art des Aufrufs:
Typ  der Szene: f                f ... Funktion
                                 k ... Kommando

Return-Wert | Folgeszene
------------|-----------
     01     | sz31
     02     | sz32
     03     | sz33
            |
            |
            |
```

```
S Z E N A R I U M pflegen          ? fordert Hilfe an
  Name: szenar1                    * aktiviert das Service

Funktionsauswahl                 Service - Funktionen

1 Loeschen eines AF-Blocks       1 Anzeige der Std-rc-Tabell
2 Doppeln eines AF-Blocks        2 Anzeige START-AF-Block
3 Akt. der Std-rc-Tabelle        3 Anzeige der AF-Bloecke
4 Akt. des START-AF-Blocks       4 Anzeige eines AF-Blocks
5 Akt. eines AF-Blocks           5 Sytemruf
6 Hinzuf. eines AF-Blocks        6 Dialogfuehrung
7 Aktualisierung beenden

*
```

```
S Z E N A R I U M  szenar1

Szene          Typ
  rc  Folgeszene  kz
  rc  Folgeszene  kz

Typ
 f ...  Funktion
 k ... Kommando

Kz
- ... Szene ist nicht im
      Szenarium enthalten
+ ... Szene wurde im Szena-
      rium nicht angesprochen

  Standard-rc-Tabelle

    !!  ENDE
     !  RETURN
    ¤¤  sz0

  START-Szene
  sz0        f
    01  sz1
    02  sz2
    03  sz3

  sz1        f
    01  sz11
    02  sz12

  sz2        k
    01  sz21

  sz11       f
    01  sz0
    02  sz1

  ....       .
    ..  ...
```

```
S Z E N A R I U M  szenar1

   Zeichenerklaerung
   -----------------

 -  ... Szene  ist  nicht weiter
         aufzuloesen
 *  ... Szene wurde weiter oben
         bereits aufgeloest
-R- ... Rekursiver Szenenaufruf

Standard-rc-Tabelle

 rc  Folgeszene

 !!  ENDE
  !  RETURN
 ¤¤  sz0
```

```
sz0
 |
 01|__sz1
 |   |
 |   01|__sz11
 |   |   |
 |   |   01|__sz0    -R-
 |   |   |
 |   |   02|__sz1    -R-
 |   |
 |   02|__sz12
 |       |
 |       01|__sz121   -
 |
 02|__sz2
 |   |
 |   01|__sz21
 |       |
 .       .
 .       .
 .       .
```

Routinen ist nun nicht mehr Bestandteil des Algorithmus, sondern wird in einer externen Tabelle, dem Szenarium, beschrieben und von einem Steuermodul, dem Manager, ausgeführt. Die Konzentration der Programmsteuerung in Tabellenform ermöglicht eine dialoggestützte Generierung und Protokollierung sowie die maschinelle Prüfung und dynamische Anpassung. Die Speicherung des Szenariums als Datei löst den physischen Zusammenhang zwischen dem Programmtext und der Programmsteuerung. Ein auf diese Weise gesteuertes Programm ist dann nicht mehr an eine vorgegebene Steuerung gebunden, denn die Datei der aktuellen Steuerung kann dem Programm zum Zeitpunkt der Abarbeitung zugeordnet werden. Jede Programmeinheit, die folgenden Bedingungen genügt, kann als Szene aufgerufen werden:

– Aufruf mit einem eindeutigen Namen
– Rückkehr zur rufenden Programmeinheit
– Rückgabe einer begrenzten Anzahl definierter return-Werte.

Diese Eigenschaften haben im allgemeinen

– Subroutinen (Unterprogramme, Funktionen, Moduln),
– Programme (Kommandos) und
– Prozeduren.

Szenarium

Die Generierung und Aktualisierung eines Szenariums erfolgt im Dialog. Für jede Szene wird die Folgesteuerung in Abhängigkeit von den return-Werten definiert. Help- und Servicefunktionen geben zusätzliche Informationen über mögliche Eintragungen und den Stand der Generierung (Abb. 1 und 2). Die Darstellung des Szenariums kann tabellarisch oder grafisch ausgeführt werden (Abb. 3 und 4). Eine wahlweise Generierung von Pseudoszenen ermöglicht das separate Testen des Szenariums. Hierdurch kann die Entwicklung der Programmsteuerung von der Programmierung bzw. Generierung der Szenen getrennt werden. Die Pseudoszenen können später schrittweise ausgetauscht werden.

Mit einer Konsistenzprüfung hat der Anwender zusätzliche Möglichkeiten, eventuelle Unvollkommenheiten und Widersprüche der Folgesteuerung aufzudecken.

Manager

Der Manager, der zentrale Steuermodul, ist die main-Funktion des Dialogprogramms. Die als START-Szene gekennzeichnete Szene wird vom Manager als erste Szene aufgerufen. Der Rückkehrcode der Szene und die Angaben des Szenariums bestimmen den Aufruf der nächsten Szene usw. Wenn die Tabelle return-Wert/Folgeszene keine Eintragung für den von der Szene gelieferten return-Wert enthält, wird die Tabelle der Standard-return-Werte ausgewertet. So ist es möglich, die speziellen return-Tabellen von oft wiederkehrenden Steuerfunktionen zu entlasten. In Abbildung 3 enthält die Standardtabelle z. B. Eintragungen für den Aufruf der vorangegangenen Szene (RETURN), den Aufruf der START-Szene sz0 und das Programmende.

Durch wiederholte Benutzung der RETURN-Steuerfunktion kann ein Dialog zurückgeblättert werden.

Der Aufruf einer Szene erfolgt in Abhängigkeit vom Typ der

```
S Z E N E N - G E N E R I E R U N G     Name der Szene: sz1
                                        Name der Maske: b1

  Feldpruefung                          ? fordert Hilfe an
                                        * aktiviert die Serviceroutine
Feldname | Pruefung                     ------------------------------
---------|-----------------------------  Pruefung von Maskenfeldern
f11      | sym()                        Standardpruefroutinen:
f13      | lim("1","10")                pfd() - Test auf Pfadnamen
f15      | lim("1",¤maxv)               sym() - Test auf Symbol
f1a      | rp1a(                        num() - Test auf numerisch
f1b      | ?                            txt(parm) -   Vergleich Feldin-
                                                     halt - Parameter
                                        lim(par1,par2) - Grenzwerttest
                                        lst("aaa"[,"aaa",...]) - Ver-
                                                gleich mit Listenwerten
                                        name( - Nutzerpruefroutine
```

Abb. 3 Tabellarische Darstellung des Szenariums
Abb. 4 Grafische Darstellung des Szenariums
Abb. 5 Dialogbild zur Definition der Prüfung von Feldern (rechts ein Helptext – gekürzt)

Szene als Funktions- bzw. als Kommandoaufruf. Liefert eine Szene einen nicht vereinbarten Rückkehrcode oder läßt sich die Folgeszene nicht aufrufen (Testphase), werden Fehlernachrichten ausgegeben und es wird die letzte Szene erneut aufgerufen.

Szenengenerierung

Ein Dialogprogramm für die Kommunikation Programm-Anwender besteht zum überwiegenden Teil aus Szenen zur Anforderung und Entgegennahme von Daten oder zur Anzeige von Informationen. Für die typische Bildschirmszene, die sich in die Abschnitte
– Modifikation der Bildschirmmaske vor der Ausgabe,
– Anzeige der Bildschirmmaske,
– Einlesen der ungeschützten Felder (Dateneingabefelder),
– Prüfung der Daten (ggf. Fehlermeldung und Wiederholung der Datenanforderung),
– Verarbeiten der Daten,
– Bildung des Rückkehrcodes
gliedert, kann eine Generierung im Dialog erfolgen (Abb. 5 und 6).
Wenn die angebotenen Möglichkeiten nicht ausreichen, kann der Anwender eine eigene Routine anschließen (Abb. 6 Feld f14 und Help-Text).
Die Generierung setzt das Benutzen des Maskengenerators MGEN /1/ voraus. Die Generierung kann als Erweiterung der dort für die Maskendefinition gebotenen Möglichkeiten angesehen werden.

Variablenarchiv

Die Datenkommunikation zwischen den Szenen kann über globale Daten oder Archivvariable erfolgen. Auch Aufrufargumente werden in dieser Form übergeben.
Archivvariable werden im Programm ähnlich wie Variable gehandhabt. Ihr Gültigkeitsbereich ist aber nicht auf ein Programm beschränkt. Die von einer Zugriffsroutine in einer Datei, dem Variablenarchiv, gespeicherten Daten können Informationen zwischen Programmen bzw. von einem Programmlauf zu einem folgenden Programmlauf übertragen:
– Datenübergabe Dialogprogramm – Verarbeitungsprogramm
– Datenaustausch zwischen Teildialogen
– reduzierende Dialogprogramme.
Das Variablenarchiv kann mehrdimensional angelegt werden, so daß eine separate Speicherung der Daten verschiedener Dialogbenutzer möglich ist. Ein zweidimensionales Archiv läßt

```
S Z E N E N - G E N E R I E R U N G     Name der Szene: sz1
                                       Name der Maske: b1

  Feldverarbeitung                    | ? fordert Hilfe an
                                      | * aktiviert die Serviceroutine
Feldname | Verarbeitung               |-------------------------------
---------|-----------------           | Verarbeitung von Maskenfeldern
f11      | ¤vname                     |
f12      | anz                        | ¤aaa - Inhalt des Maskenfeldes
f14      | rv14(                      |        wird als Archivvar. gesp.
f17      | ?                          | aaa  - Inhalt des Maskenfeldes
                                      |        wird als glob. Var. gesp.
                                      | aaa( - Verarbeitung des Feldes
                                      |        durch eine Nutzerrout.
```

Abb. 6 Dialogbild zur Definition der Verarbeitung von Feldern (rechts ein Helptext – gekürzt)

für jeden Benutzer die unabhängige Verwaltung von Dialogvarianten zu usw. Das Variablenarchiv kann bis zu vier orthogonale Dimensionen verwalten. Der Dateizugriff erfolgt mit nur zwei bis drei E/A-Zugriffen (Hash-Verfahren), die Speicherung ist reorganisationsfrei.

Anwendung

Mit UDIS können die Steuerung und ein großer Teil der Bildschirmszenen für ein Dialogprogramm generiert werden. Es gibt praktisch keine Einschränkungen bezüglich der Struktur der externen Steuerung (Hierarchie, Netzstruktur, Rekursionen, Kombinationen), der Szenenanzahl, der Anzahl der return-Werte pro Szene, der Anzahl der Maskenfelder usw.
Große Dialoge können in Teildialoge untergliedert und jeder Teildialog als eigenständiger Dialog entwickelt und später als Szene vom Typ *Kommando* zu einem Gesamtdialog montiert werden. Die freie Zuordnung des Szenariums bzw. des Variablenarchivs als Dateien ermöglicht die Arbeit mit unterschiedlichen Steuerungen und mehreren Varianten.
UDIS bietet für die Programmerarbeitung, das Testen und Aktualisieren auch dann Vorteile, wenn es sich nicht um ein ausgesprochen bildschirmorientiertes Dialogprogramm handelt. Parallelarbeit an umfangreichen Programmen/Programmsystemen ist möglich durch
– Trennung Steuerung/Algorithmus (Szenarium/Szenen),
– Separieren des Algorithmus in Szenen, die bei fehlender oder geringer Datenkopplung als selbständig zu betrachten sind,
– Aufteilung des Programms in Teilprogramme mit der Möglichkeit von Montage und Demontage,
– stufenweises Austesten der einzelnen Teilprodukte.
Die genannte Arbeitsweise unterstützt die Lokalisation von Fehlern sowohl in der Phase der Programmierung und des Testes als auch während der Erprobung.
Eine interessante Anwendung ist der Aufruf von Verarbeitungsprogrammen eines Projekts. Es ergibt sich eine Programmfolgesteuerung, die der herkömmlichen Jobsteuerung entspricht.
Die Hauptkomponenten des Dialogsystems externe Programmsteuerung (Szenarium), Szenengenerierung und Variablenarchiv sind auch unabhängig voneinander einsetzbar. Die Szenengenerierung kann als Ergänzung des Maskengenerators MGEN angesehen werden. Das Variablenarchiv hat die Eigenschaften einer allgemeinen Zugriffsmethode.

Literatur
/1/ Blacha, B.: MGEN – Ein Maskengenerator
rechentechnik/datenverarbeitung
3/89

SOFTWARE-TECHNOLOGIE

LS2 – Softwaremonitor und Steuersystem für SVM

Dr. Eckhart Richter
VEB Rechenzentrum
Karl-Marx-Stadt

Im praktischen Rechenbetrieb über Multiprogrammbedingungen ist es notwendig, sowohl die Nutzung der Hardwareressourcen als auch den gebotenen Service regelmäßig zu kontrollieren und im Sinne der Betriebsziele der entsprechenden Installation zu beeinflussen. Das hier vorzustellende Programmsystem LS2 (System zur Steuerung von Leistung und Service) unterstützt für das Betriebssystem SVM/ES weitgehend diese Aufgabe.
Das Programmsystem LS2 ist lauffähig auf ESER-Rechnern der Reihen 2 und 3 unter den Betriebssystemen SVM-3.3K1 und SVM-3.5 sowie auf Rechnern mit IBM/370-Architektur ab VM/SP Rel.3.
LS2 ist ein leistungsfähiger Softwaremonitor mit zusätzlichen Steuerungsfunktionen. Als Monitor erlaubt LS2 sowohl die operative Anzeige von Meßwerten als auch postoperative Analysen nach verschiedenen Gesichtspunkten und mit verschiedenen Formen der Ergebnisdarstellung. Es werden Kenngrößen zum Bewerten der Hardwarenutzung und der den Nutzern gebotenen Servicebedingungen geführt.
Die ausgewiesenen Servicekenngrößen unterstützen die Vereinbarung und Kontrolle definierter Servicebedingungen zwischen Betreiber und Nutzer von Rechnersystemen. Der Steuerungsanteil von LS2 übernimmt die Sicherung vereinbarter Servicebedingungen mit Ressourcen-Steuerkommandos (z. B. SET PRIORITY oder SET FAVOR) des Betriebssystems entsprechend eines vorgebbaren Serviceplanes und einer Steuerstrategie.
Das Programmsystem LS2 wurde sorgfältig an die Erfordernisse des praktischen Rechenbetrie-

bes angepaßt, was sich u. a. in folgenden Qualitätsmerkmalen widerspiegelt:
– Der Eigenbedarf an Rechnerressourcen für Messung und Steuerung ist minimal (etwa 1 Prozent der ZVE-Leistung).
– Der für die LS2-Datenbasis benötigte Plattenspeicherbereich wird sparsam genutzt. Für die Speicherung der Meßdaten über zwei Monate sind etwa 10 Zylinder eines 100-MByte-Plattenstapels erforderlich.
– Die LS2-Datenbasis ist wartungsfrei. Tagesdateien können auf Monatsdateien kumuliert und nach einer Verfallsfrist gelöscht werden.
– Alle Komponenten präsentieren sich mit einer einheitlich gestalteten Nutzerschnittstelle in Form von Menüs, Panelen, Helps sowie PF-Tasten für spezielle Funktionen.
– Alle Laufsituationen von der Gegenwart bis etwa zwei Monate zurück sind durch Anzeige der aufgezeichneten Meßwerte jederzeit unmittelbar reproduzierbar.

Prinzipielle Wirkungsweise

Das Programmsystem LS2 besteht aus mehreren Komponenten, deren Zusammenwirken in Verbindung mit den entsprechenden Dateien auf der 4. US (A) dargestellt ist.
Die Meß- und Steuerkomponente als Herzstück des Programmsystems läuft in einer Steuermaschine im DISCONNECTED-Zustand. In dieser Maschine werden die Meßwerte ermittelt, in die Datenbasis eingetragen, an die Analysemaschinen geschickt sowie Steuermaßnahmen ermittelt und ausgeführt. Das Meßverfahren beruht auf der Differenzbildung von Zeitsummen- und Ereignissummenwerten. Im Takt des Meßintervalls werden aus den Steuerblöcken des Betriebssystems die gewünschten Summenwerte abgegriffen und ihre Differenz zu den zur letzten Messung erhaltenen Summenwerten gebildet. Dieses Meßverfahren garantiert einen zu vernachlässigenden Meßaufwand und ist damit für den Dauerbetrieb sehr gut geeignet. Die ermittelten Meßwerte werden für postoperative Analysen in die LS2-Datenbasis gespeichert und können außerdem direkt am Bildschirm einer Analysemaschine eingesehen werden.
Die Analysekomponente ist gleichzeitig in mehreren virtuellen Maschinen nutzbar, um Monitordaten operativ oder postoperativ zu analysieren. Die Übertragung der Meßwerte von der Steuermaschine zu einer Analysemaschine zum Zweck der operativen Anzeige erfolgt über die Kommunikationskomponente IUCV des Betriebssystems.
Mit Hilfe einer Planungskomponente kann für jeden Wochentag ein Serviceplan erstellt werden, der der Steuerkomponente als Basis für Steuermaßnahmen dient. Über eine Einstellkomponente ist LS2 an die speziellen Bedingungen jeder Installation anpaßbar. Eine Informationskomponente gibt dem Dialognutzer Auskunft über den Serviceplan.

Kategorie	Kenngröße	Gesamtsystem	virt. Maschinen	Servicegruppen	Bedeutung
Leistung	L	x	x	x	Leistung
	CPUT	x	x	x	ZVE-Nutzungsgrad gesamt
	CPUV	x	x	x	ZVE-Nutzungsgrad virtuell
	PAGR	x	x	x	Seitenwechselrate
	SIOR	x	x	x	E/A-Rate
	WS		x	x	Arbeitsseitenmenge
Service	S	x	x	x	Service
	EXP	x	x	x	Verzögerungsfaktor
	CRT	x	x	x	Antwortzeit auf einfache Interaktionen
Last	MTF	x			Multitransaktionsfaktor
	MZF	x			Multizuordnungsfaktor
	NACT	x		x	Anzahl aktiver Maschinen
	NDIA	x		x	Anzahl aktiver Dialogmaschinen
	LOG	x		x	Anzahl LOGON-Maschinen
Behinderung	IDLW	x			Wartegrad auf Arbeit
	PAGW	x			Wartegrad auf Paging
	SIOW	x		x	Wartegrad auf E/A-Ende
Nutzerproduktivität	IUP	x	x	x	Zahl der Eingaben pro Nutzer pro Minute
Steuerung	STATE	x			Steuerzustand

$$L = \left(GCPU \frac{CPUT}{MCPU} + GPAG \frac{PAGR}{MPAG} + GSIO \frac{SIOR}{MSIO} \right) * 100\%$$

$$S = \frac{1}{GCRT \frac{CRT}{MCRT} + GEXP \frac{EXP}{MEXP * LF}}$$

CPUT, PAGR, SIOR, CRT, EXP sind Meßwerte (s.a. Tabelle 1).
GCPU, GPAG, GSIO, GCRT, GEXP sind Gewichtsfaktoren.
MCPU, MPAG, MSIO, MCRT, MEXP sind normative Werte.
LF ist ein Leistungsfaktor und bestimmt sich aus der Leistungsfähigkeit des Rechnersystems.

Tab. Von LS2 erfaßte Kenngrößen
Abb. 1 Synthetische Kenngrößen L und S

Kenngrößen

In der Tabelle sind die von LS2 erfaßten Meßwerte, sortiert nach Bewertungskategorien aufgeführt. Da über die Kategorien Last, Leistung und Service und über die zu ihrer Bewertung verwendeten Kenngrößen unterschiedliche Vorstellungen existieren, wird nachfolgend die bei LS2 hinterlegte Bedeutung erläutert.
In Anlehnung an den physikalischen Leistungsbegriff wird unter *Rechnerleistung,* hier kurz als Leistung bezeichnet, die in einer Beobachtungszeit erbrachte Rechnerarbeit, dividiert durch die Beobachtungszeit, verstanden. Rechnerarbeit wird immer in Realisierung einer Transaktion an einer Hardwareressource geleistet. Ein Maß für die Rechnerarbeit an der Ressource ZVE ist die Belegungszeit der ZVE durch die entsprechende Transaktion. Die am E/A-System geleistete Arbeit entspricht dem Umfang der übertragenen Daten. LS2 bietet als Maß für E/A-Arbeit die Anzahl der ausgeführten E/A-Operationen an, da der tatsächliche Umfang der übertragenen Daten derzeit mit vertretbarem Aufwand in der SVM-Umgebung nicht meßbar ist.
Mit *Service* bzw. Serviceniveau soll ein Aspekt der Qualität erbrachter Rechnerleistung bezeichnet werden. Qualitätskriterium ist dabei, ob die bereitgestellte Rechnerleistung den Anforderungen der Nutzer entspricht, d. h. diese ihre Aufgaben erfüllen können, oder nicht. Die Beurteilung des Serviceniveaus ist somit nicht zuletzt von der Erwartungshaltung der Nutzer abhängig. Zur quantitativen Fassung der Kategorie Service dienen bei LS2 die Kenngrößen Antwortzeit und Verzögerung.
Im Multiprogrammbetrieb werden zeitgeteilt mehrere Transaktionen quasi-parallel realisiert. Das führt insbesondere an den Ressourcen ZVE, Realspeicher und E/A-System zur Bildung von Warteschlangen. Unter *Last* oder Belastung an einer Ressource wird die an dieser Ressource anstehende Arbeitsmenge verstanden. Als Maß für die Last kommt vorzugsweise die Anzahl der anstehenden Transaktionen (Multitransaktionsfaktor) in Betracht.
Um die Kategorien Leistung und Service jeweils in eine Meßzahl fassen zu können, werden für Leistung und Service synthetische Kenngrößen (L und S) verwendet (Abb. 1).
Die Leistung, für eine virtuelle

Maschine oder für das Gesamtsystem, berechnet sich als gewichtete Summe aus den Nutzungswerten der Hauptressourcen ZVE, E/A-System und Paging-System. Der so berechnete Leistungswert entspricht der Bruttoleistung. Die Parameter sind, in Abhängigkeit vom Leistungsvermögen der Hardware und dem realen Arbeitslastprofil, so zu wählen, daß sich bei maximaler Nutzleistung ein Leistungswert von L=100 Prozent ergibt. In der Installationsphase kann mit Standardwerten gearbeitet werden. Nachdem Meßwerte des speziellen Rechnersystems mit LS2 erfaßt wurden, können die LS2-Analysemittel zum Nachstellen der Parameter verwendet werden.

Aus der in Abbildung 2 gezeigten Systemanalyse kann der Lastbereich mit maximaler Nutzleistung des untersuchten Rechnersystems abgelesen werden. Bezüglich der ZVE-Nutzung liegt der günstigste Bereich bei Multitransaktionsfaktoren zwischen 3,5 und 5,0. Die maximale Nutzung des E/A-Systems stellt sich im Mittel bei Multitransaktionsfaktoren zwischen 2,0 und 4,0 ein.

Der Service wird analog zur Leistung aus der gewichteten Summe der Einflußgrößen Antwortzeit und Verzögerungsfaktor ermittelt. Die Parameter sind so zu wählen, daß ein Servicewert S = 1 dem mit dem Nutzer vereinbarten Serviceniveau entspricht.

Der Verzögerungsfaktor ist definiert als das Verhältnis der Antwortzeit für eine Transaktion im Multiprogrammbetrieb, dividiert durch die Antwortzeit im Alleinlaufbetrieb. Der Verzögerungsfaktor ist damit ein gutes Maß zum Bewerten des Serviceverhaltens des Rechnersystems bezüglich der Stapelaufgaben (einschließlich lang laufender Dialogaufgaben).

Die Antwortzeit CRT ist die mittlere Antwortzeit auf einfache Interaktionen (Trivial Response Time). Unter einer einfachen Interaktion wird eine Interaktion verstanden, zu deren Ausführung weniger als eine ZVE-Zeitscheibe benötigt wird. In dieser Form definiert und gemessen ist CRT ein gutes Maß für das Serviceverhalten im Dialogbetrieb. In Abbildung 3 sind Leistung, Service und Nutzerproduktivität in Abhängigkeit von der Last (MTF) dargestellt. Die Nutzerproduktivität ist mit etwa 13 Interaktionen pro Minute pro Nutzer erwartungsgemäß bei MTF = 1 am größten. Beim Betrieb des Rechners im Bereich maximaler Nutzleistung beträgt die Nutzerproduktivität nur noch etwa neun Interaktionen pro Nutzer pro Minute (Verringerung um etwa 30 Prozent). Bei unkontrollierter Überlastung des Rechners verringert sich die Nutzerproduktivität weiter und zugleich sinkt die Nutzleistung.

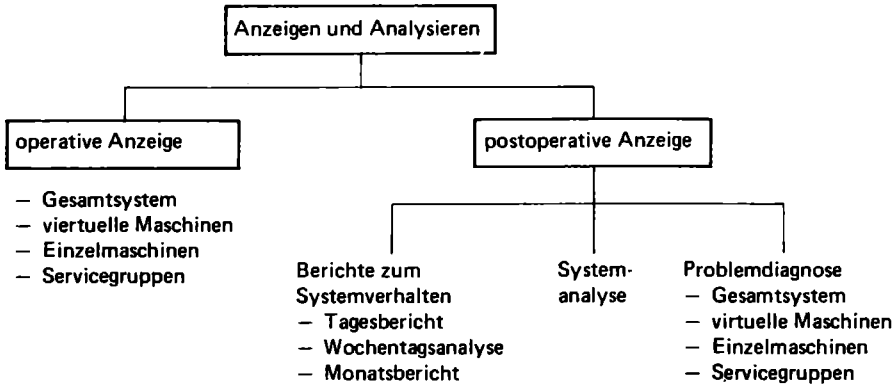
```
LSA 3.1                  - SYSTEMANALYSE -                  SYSTEM: XYZ
ANALYSEZEITRAUM: 20.07.88 - 25.09.88, VON 07 - 17 UHR, INSGESAMT 25752 MIN.

        KURVEN SIND GEGLAETTET
MTF                                                          L      S    IUP
-----0----1----2----3----4----5----6----7----8----9----0 ------ ---- -----
 0.0 | *                        Q   +                    |   9.33 2.95  9.46
 0.5 |      *                  +      O                  |  28.21 2.30 13.47
 1.0 |           *         +           O                 |  46.42 1.99 12.06
 1.5 |                *+             O                   |  66.79 1.72 11.66
 2.0 |              +     *       O                      |  81.69 1.23 10.49
 2.5 |            +          *  O                        |  89.81 1.04  9.79
 3.0 |          +              *O                        |  94.94 0.93 10.28
 3.5 |        +                O*                        | 101.19 0.81  9.24
 4.0 |      +                O   *                       | 107.38 0.70  8.65
 4.5 |      +              O       *                     | 113.06 0.62  9.06
 5.0 |     +             O          *                    | 106.94 0.59  7.73
 5.5 |    +            O            *                   | 112.77 0.51  7.63
 6.0 |    +          O                *                  | 122.38 0.47  7.33
 6.5 |   +         O                    *                | 119.10 0.38  5.70
 7.0 |   +       O                        *              | 111.31 0.42  6.64
 7.5 |   +     O                            *            | 128.78 0.34  6.14
 8.0 |  +    O                               *           | 156.33 0.23  4.37
 8.5 |  +  O                                   *         | 135.00 0.23  5.53
 9.0 | + O                                       *       | 121.00 0.10  2.10
 9.5 | +O                                          *     | 155.00 0.20  3.20
10.0 | +O                                            *   | 174.00 0.20  4.20
-----0----1----2----3----4----5----6----7----8----9----0 ------ ----- ----

  KENNGROESSE SYMBOL SKALIERUNG STREUUNG          BEDEUTUNG
-------------------------------------------------------------------------
   MTF                                  MULTITRANSAKTIONSFAKTOR
   L          *        0.05             LEISTUNG GESAMTSYSTEM IN %
   S          +        2.00             SERVICE
   IUP        O        0.50             INTERAKTIONEN PRO NUTZER PRO MINUTE
```

```
LSA 3.1                  - SYSTEMANALYSE -                  SYSTEM: XYZ
ANALYSEZEITRAUM: 12.06.88 - 25.09.88, VON 00 - 24 UHR, INSGESAMT 60888 MIN.

        KURVEN SIND GEGLAETTET
MTF                                                        CPUT  CPUV  SIOR
-----0----1----2----3----4----5----6----7----8----9----0 ----- ----- -----
 0.0 | O  *                                             |  8.79  4.90  2.26
 0.5 |      O +    *                                    | 31.56 19.01  8.38
 1.0 |            O+     *                              | 51.91 34.42 16.18
 1.5 |                 + O       *                      | 62.81 41.24 22.43
 2.0 |                    +    O       *                | 71.25 45.74 30.43
 2.5 |                      +    O        *             | 77.98 49.73 29.06
 3.0 |                        +    O          *         | 83.65 53.67 31.30
 3.5 |                          +   O            *      | 86.90 55.08 29.88
 4.0 |                            + O              *    | 88.75 54.86 30.49
 4.5 |                            +O                *  | 88.87 55.85 28.70
 5.0 |                            O                  *  | 94.65 56.67 27.07
 5.5 |                          O                     *  | 93.98 53.93 25.78
 6.0 |                         O+                     *  | 93.16 51.58 26.34
 6.5 |                        O+                      *  | 94.26 49.83 23.20
 7.0 |                       O +                      *  | 97.39 58.50 25.56
 7.5 |                      O +                       *  | 95.36 50.82 20.36
 8.0 |                      O+                        *  | 84.40 39.00 21.40
 8.5 |                     O+                          * | 85.50 37.00 25.50
 9.0 |                    +O                           * | 98.50 37.50 18.00
 9.5 |                   + O                           * | 90.00 34.00 25.00
10.0 |                 +   O                           * | 96.00 40.00 24.00
-----0----1----2----3----4----5----6----7----8----9----0 ----- ----- -----

  KENNGROESSE SYMBOL SKALIERUNG STREUUNG          BEDEUTUNG
-------------------------------------------------------------------------
   MTF                                  MULTITRANSAKTIONSFAKTOR
   CPUT       *        0.10             ZVE-NUTZUNG GESAMT IN %
   CPUV       +        0.10             ZVE-NUTZUNG VIRTUELL IN %
   SIOR       O        0.20             E/A-RATE PRO SEKUNDE
```

Abb. 2 Bestimmung des Bereichs der maximalen Nutzleistung
Abb. 3 Lastabhängigkeit von Leistung, Service und Nutzerproduktivität

Anzeigen und Analysieren

Die mit LS2 möglichen Anzeige- und Analysefunktionen sind in Abbildung 4 dargestellt. Die operative Anzeige erlaubt die unmittelbare Verfolgung des Laufverhaltens des Rechnersystems (Real-Time-Monitor). Angeboten wird die Anzeige des Gesamtsystems, aller virtueller Maschinen, einzelner virtueller Maschinen oder von Servicegruppen. Zur Anzeige kommt eine feste Menge von Kenngrößen in chronologisch-tabellarischer Form am Bildschirmterminal. Ein Beispiel für die operative Anzeige zeigt Abbildung 5 (CRT-Werte sind in 0,1 Sekunden ausgewiesen).



LSA 1.1 OPERATIVE ANZEIGE Gesamtsystem System: XYZ
Datum: 14.08.87

BEGINN	SEK	L	S	CPUT	CPUV	IDLW	PAGW	SIOW	PAGR	SIOR	MTF	MZF	EXP	CRT
12:58:09	179	71	2.1	82	53	0	2	16	2	38	1.9	1.9	1.7	3
13:01:08	180	58	2.5	61	29	0	2	37	2	33	1.7	1.7	2.0	1
13:04:08	180	67	3.2	60	29	0	2	38	6	32	1.3	1.3	1.3	1
13:07:08	180	62	2.6	60	28	0	2	39	5	31	1.5	1.5	1.8	1
13:10:08	180	72	1.7	66	32	0	1	33	6	36	1.8	1.8	2.0	3
13:13:08	180	72	1.4	61	27	0	2	37	10	28	2.0	2.0	2.6	3
13:16:08	180	89	1.6	68	30	0	1	31	10	45	2.0	2.0	1.9	4
13:19:08	180	74	1.7	61	27	0	2	37	10	31	1.6	1.6	1.8	3

PF: 1=HELP 3=END 4=BEFORE 5=LATER 12=HOME

Abb. 4 Hierarchie der Analysefunktionen
Abb. 5 Operative Anzeige des Gesamtsystems

Mit der Funktion Berichte zum Systemverhalten werden Analysen über bestimmte Zeiträume angefertigt. Die Darstellung erfolgt in quasigraphischer oder graphischer Form wie das Beispiel auf der 4. US (B) zeigt. Die gewünschten Kenngrößen, Zeiträume und Maschinen bzw. Maschinengruppen sind in einem Eingabepanel wählbar. Die grafische Anzeige erfordert entsprechende Gerätetechnik, z. B. Bildschirmterminals IBM 3279. Das Beispiel in B zeigt Leistung und Service für eine Nutzergruppe (alle Maschinen, die mit dem Maschinenkennzeichen PROG beginnen) für einen Tag. Als Liniendiagramm ist der Steuerzustand hinterlegt.

Die auf der 4. US (C) gezeigte Wochentagsanalyse für Donnerstag über einen Zeitraum von anderthalb Monaten gibt Auskunft über die Mittelwerte der Servicekenngrößen Antwortzeit und Verzögerungsfaktor. Der daraus errechnete synthetische Servicewert S liegt im Mittel über 1, d. h. das vereinbarte Serviceniveau wurde erreicht.

Die Funktion Systemanalyse dient der Analyse von Abhängigkeiten zwischen Kenngrößen über wählbare Zeiträume. Grundlage für die Analyse sind die in der Datenbasis gespeicherten Meßwerte. Um die Zusammenhänge zwischen den Kenngrößen deutlicher zu machen, können die Kurven geglättet werden (Approximation mit kubischen Spline-Funktionen, siehe auch Abb. 2 und 3).

Die Funktion Problemdiagnose erlaubt die nachträgliche, postoperative Verfolgung von Laufsituationen mit dem Ziel, Problemsituationen zu erkennen, ihre Ursachen zu ermitteln und gegebenenfalls Systemverbesserungen daraus abzuleiten. Die Präsentation der Ergebnisse ist weitgehend analog zur operativen Anzeige.

Steuern

Die Steuerung von Leistung und Service mit LS2 beruht auf der Beeinflussung der Ressourcennutzung einzelner virtueller Maschinen mittels Änderung der Auswahlprioritäten dieser Maschinen. Dafür werden vorrangig die Betriebssystemkommandos SET PRIORITY, SET LIMIT und SET FAVOR verwendet.

Welche virtuellen Maschinen zu beeinflussen sind, beschreibt ein Serviceplan. Ein solcher Plan existiert für jeden Wochentag und in ihm können zeitscheibenweise die Servicebedingungen für einzelne virtuelle Maschinen und für Maschinengruppen festgelegt werden. Mit dem Serviceplan ist zu jedem Zeitpunkt die Einteilung aller aktiven virtuellen Maschinen in drei Servicegruppen mit folgender Bedeutung gegeben:

Servicegruppe 1 – guter und sehr guter Service
Servicegruppe 2 – durchschnittlicher Service
Servicegruppe 3 – verringerter Service bei starker Rechnerlast.

Alle im Plan nicht aufgeführten Maschinen werden der Servicegruppe 4 zugeordnet.

Darüber hinaus können für Maschinen der Servicegruppe 1 Sonderbedingungen geplant werden, die eine besondere Bevorzugung in der Ressourcennutzung zur Folge haben (Favorisierung, residente Seiten, Reservierung von Seitenrahmen).

Die derzeit wichtigsten Steuerziele sind:
– Betrieb des Rechnersystems im Bereich maximaler Nutzleistung und
– Sicherung eines vereinbarten Serviceniveaus für Nutzer der Servicegruppe 1.

Inwieweit das zweite Steuerziel durch LS2 realisiert werden kann, zeigt die grafische Darstellung auf der 4. US (D). Hier sind die Servicewerte für alle drei Servicegruppen in Abhängigkeit von der Rechnerlast dargestellt. Das vereinbarte Steuerziel S = 1 für Nutzer der Servicegruppe 1 kann im Mittel bis zu einem Lastwert von MTF = 6 gehalten werden. Das Programmsystem LS2 wird seit Februar 1988 vertrieben. Die Unterstützung des Doppelprozessorsystems EC 1057 erfolgt durch eine Version 2, die ab September 1989 verfügbar sein wird. Längerfristig wird, aufbauend auf den bisherigen Erfahrungen und Ergebnissen, an der Entwicklung eines wissensbasierten Systems zur Leistungsbewertung gearbeitet. Ein solches System soll Spezialisten bei der Analyse und Interpretation von Meßwerten unterstützen.

VERANSTALTUNGEN

Ausstellung zum Produktionsprogramm der Firma VIDEOTON

Termin: 2. Hälfte April 1989
Ort: Berlin
Veranstalter:
Videoton AG Vertretung bei der Botschaft der UVR
Schwerpunkte:
Vorführung moderner Rechentechnik, Anwendung leistungsfähiger Personalcomputer, Möglichkeiten zur Schaffung lokaler Netze, Robotertechnik, Unterhaltungselektronik
Organisatorische Hinweise:
Da bis Redaktionsschluß der genaue Ausstellungstermin noch nicht feststand, können die exakten Daten über den technischen Kundendienst der Videoton AG Vertretung, Arnold-Zweig-Str. 11, Berlin, Pankow 1100, Telefon: 472 41 80 (185, 186) eingeholt werden

2. Internationale Fachtagung Lichttechnik

Termin: I. Quartal 1990
Ort: Berlin
Veranstalter:
Fachverband Elektrotechnik in der Kammer der Technik, Wissenschaftliche Sektion Lichttechnik Kombinat VEB NARVA
Schwerpunkte:
– Aufgaben der Licht-, Ultraviolett-, Infrarot-Beleuchtungs- und Bestrahlungstechnik (neue Entwicklungen, rationelle Energieanwendung)
– Einsatz der Mikroelektronik und Rechentechnik bei der Erzeugnisentwicklung, insbesondere bei Projektierungsaufgaben (CAD/CAM)
– Einsatz der optischen Strahlungstechnik für Hochtechnologien (Mikroelektronik, Biotechnologie, Medizin u. a.)
– Neueste Erkenntnisse der Grundlagenforschung
Organisatorische Hinweise:
Zu diesen Themen wird um Vorschläge für Vorträge und Poster bis zum 31. Mai 1989 gebeten. Teilnahmemeldungen sind schriftlich an folgende Anschrift zu richten: Kammer der Technik – Präsidium – Fachverband Elektrotechnik, Clara-Zetkin-Str. 115/117, Berlin, 1086

Dialogorientiertes Auskunftssystem für ESER

Claus Dieter Krah,
Roland Baum,
Hartmut Schettler,
Klaus Ginter
VEB Bandstahlkombinat „Hermann Matern", Eisenhüttenstadt Ost, Bereich ORZ

Mit zunehmender Integration der Informationsverarbeitungsprozesse steigt der Bedarf, die in den Datenverarbeitungszentren vorhandenen umfangreichen sequentiellen Dateien über anwenderfreundliche Datenbanksysteme für temporäre Auskunftssysteme zu erschließen. Mit dem im Eisenhüttenkombinat Ost projektierten universellen Auskunftssystem (UAS) wird dem Anwender ein Datenbankprojekt zur Verfügung gestellt, mit dessen Hilfe Auskünfte im Dialog ohne vorherigen Programmieraufwand zusammengestellt werden können. Die Auswahl der Daten, die gewünschten logischen Verknüpfungen und die bildschirmgerechte Aufbereitung erfolgt mit Kommandos anhand einmalig vorher eingespeicherter Tabellen. Die Tabellentechnik ist einfach und übersichtlich gestaltet. Das Einspeichern der ausgewählten sequentiellen Dateien in die Datenbank erfolgt im Jobbetrieb.
Das Auskunftssystem UAS arbeitet unter Steuerung der Teilnehmerunterstützung TSO im Betriebssystem OS (TSO) auf EDVA vom Typ ESER und benötigt 300 KByte Hauptspeicher.

Leistungsvermögen

Das Universelle Auskunftssystem UAS beinhaltet
– die Generierung (Erstellung) der Datenbank
– das Einspeichern der benötigten Tabellen zum Beschreiben der einzelnen Dateistrukturen, der abzuarbeitenden Kommandos einschließlich der Parameterauswertung sowie der auf den Terminals auszugebenen Bilder in die Datenbank
– das Einspeichern beliebiger sequentieller Dateien anhand der zugehörigen Dateibeschreibungstabelle
– das Listen der Generierungsparameter sowie des Datenbankverzeichnisses
– das Listen der eingespeicherten Tabellen
– das Löschen nicht mehr benötigter Dateien und Tabellen aus der Datenbank
– das Eröffnen des Dialogs auf dem Bildschirm mittels Kennwort
– das Erzeugen und Anzeigen von Auskunftsbildern innerhalb des Dialogs mittels Eingabe der entsprechenden Kommandos und Kommandoparameter
– das Anzeigen der möglichen Kommandos und ihre Erläuterung in einem Menü auf dem Bildschirm
– das Beenden des Dialogs.

Erstellen und Vorbereiten des Auskunftssystems für den Dialog

Generierung der UAS-Datenbank

Das Erstellen der Datenbank für das Auskunftssystem erfolgt mit einem Job. In ihm wird die Größe der Datenbank, die Anzahl und die Codeworte der Bildschirme sowie die maximale Anzahl von Auskunftsbildern, die von einem Kommando erzeugt werden können, festgelegt. Die Größe der Datenbank wird nur von der Speicherkapazität der Plattengeräte begrenzt. Maximal können 32 Bildschirme vom Auskunftssystem gesteuert und 50 Auskunftsbilder von einem Kommando erzeugt werden. Bei der Generierung wird in der Datenbank ein leeres Verzeichnis der eingespeicherten Tabellen und Dateien eingerichtet.

Aufbau der Tabellen

Für die Steuerung des Auskunftssystems sind drei Typen von Beschreibungstabellen notwendig. Alle Tabellen werden unter TSO erstellt und aktualisiert. Die Reihenfolge der notwendigen Angaben ist einzuhalten. Zwischen ihnen sind eine beliebige Anzahl von Leerzeichen, Leerzeilen oder Kommentaren möglich, welche die Tabellen übersichtlich gestalten können.
Die Übernahme der Beschreibungstabellen in die Datenbank erfolgt mit speziellen Jobs und ist, soweit sie nicht aktualisiert werden müssen, nur einmal notwendig. Eine Prüfung der Werte auf Zulässigkeit wird von den Jobs für die Tabellenaufnahme realisiert. Im Fehlerfall erhält der Anwender eine Beschreibung des Fehlers auf dem Protokoll.
– Dateibeschreibungstabellen
Für jede aufzunehmende Datei muß in der Datenbank eine Dateibeschreibungstabelle vorhanden sein. Sie definiert den Aufbau eines Datensatzes der jeweiligen Datei. In ihr werden die Suchbegriffe, welche bestimmend für die Auswahl der Datensätze in einem Auskunftskommando sind, und die zu übernehmenden Datenworte beschrieben. Im einzelnen sind folgende Parameter anzugeben:
- Name der zu beschreibenden Datei
- Satzlänge dieser Datei
- maximale Satzanzahl
- Suchbegriff- und Datenwortanzahl
- pro Suchbegriff den Namen des Suchbegriffs und die maximale Anzahl von Werten, die dieser Suchbegriff innerhalb der Datei annehmen kann
- pro Datenwort (Suchbegriffe sind gleichfalls Datenworte) deren Namen, Position im Datensatz, Datentyp und Länge.

Eine Datei kann nach maximal 20 Suchbegriffen ausgewertet werden.
– Bildbeschreibungstabellen
Für jede auszuwertende Datei muß in der Datenbank eine Bildbeschreibungstabelle vorhanden sein. Sollen die Sätze der Datei bei der Auskunft auf verschiedene Art angezeigt werden, sind mehrere Bildbeschreibungstabellen notwendig. Möglich ist ihre Aufnahme jedoch nur bei Vorhandensein der zugehörigen Dateibeschreibungstabelle. In den Bildbeschreibungstabellen wird die Struktur des auf dem Bildschirm auszugebenden Bildes vereinbart. Sie beinhalten sowohl auszugebende erläuternde Texte als auch die Positionen und Typen der anzuzeigenden Daten (Format).
Im einzelnen sind folgende Parameter anzugeben:
- Name der Bild- und der zugehörigen Dateibeschreibungstabelle
- Satzanzahl pro Bild, weitere Sätze erzeugen Fortsetzungsbilder in den Bildspeichern für den jeweiligen Bildschirm
- Überschriftformat, wird pro Bild nur einmal ausgewertet
- Anzahl und Namen der auszugebenen Datenworte einschließlich Suchbegriffe
- Satzformat, das pro Datensatz bis zum Füllen eines Bildes wiederholt ausgewertet wird.

Im Satzformat wird das auszugebende Datenwort, dessen mögliche Verknüpfung mit anderen Datenworten, die Ausgabeposition und die Ausgabeform festgelegt. Eine Verknüpfung erfolgt mit Addition, Subtraktion und Multiplikation.
Pro Bild können maximal 22 Zeilen belegt werden. Die Namen der auszugebenden Datenworte müssen mit den in der Dateibeschreibungstabelle vereinbarten Namen identisch sein.
– Kommandobeschreibungstabellen
Die Kommandobeschreibungstabellen enthalten die zulässigen Kommandos innerhalb des Dialogsystems. In ihr werden die Kommandoparameter und deren Verknüpfung untereinander vereinbart. Zu einer Datei können mehrere auswertende Kommandos existieren, jedoch kann nur eine Datei von einem Kommando behandelt werden. Die Aufbereitung der Daten für die Ausgabe erfolgt anhand einer Bildbeschreibungstabelle. Die entsprechenden Datei- und Bildbeschreibungstabellen müssen bei Aufnahme der Kommandobeschreibungstabelle bereits in der Datenbank vorhanden sein.
Folgende Angaben müssen die Tabellen enthalten:
- Name der Kommando- sowie der zugehörigen Datei- und Bildbeschreibungstabellen
- Anzahl der Kommandoparameter (diese sind bei Aufruf des Kommandos im Dialog zusammen mit dem Kommando auf Bildschirm einzugeben)
- Eingabe der Bildschirmnummern, welche dieses Kommando starten dürfen
- Logik bzw. Auswahlvorschrift. Mit Angabe der Suchbegriffe wird die Reihenfolge der Kommandoparameter festgelegt (der Dialogteilnehmer muß bei Aufruf eines Kommandos pro Parameter die zu suchenden Werte bzw. Wertintervalle angeben). Verknüpfungen der Kommandoparameter können mit logischem UND sowie logischem ODER realisiert werden
- Eingabe eines Kommando-Menüs, das auf maximal 21 Zeilen eine Kurzbeschreibung des Kommandos beinhaltet. Dieses Menü kann mittels eines speziellen Kommandos auf dem Bildschirm angesehen werden.

Einspeicherung der Dateien

Die sequentiellen Dateien werden einzeln durch einen Job in die Datenbank eingespeichert. Dateien mit kleiner Satzlänge werden geblockt. Beim Einspeichern der Dateien werden zum schnellen Wiederauffinden der benötigten Informationen Suchdateien (mehrstufige Indexdateien) angelegt. Die Größe dieser Dateien und die Anzahl der Stufen wird vom Umfang der jeweiligen Datei und von der maximalen Werteanzahl des entsprechenden Suchbegriffs bestimmt.

Eine bereits vorhandene Datei einschließlich der zugehörigen Suchdateien wird bei Aufnahme gelöscht. Die dabei frei werdenden Sektoren werden neu belegt.
Der Anwender erhält bei Aufnahme einer Datei ein Protokoll, welches die Datei und die Erstellung der einzelnen Suchdateien nachweist. Im Fehlerfall kann nach der Korrektur die Aufnahme wiederholt werden.

Sicherung der Datenbank

Für das Sichern der Datenbank auf Magnetband und das Neuerstellen anhand der Magnetbanddatei existieren die benötigten Jobs. Bei der Übernahme auf Magnetband werden nur die eingespeicherten Daten der Datenbank gedoppelt, was die Sicherungszeit erheblich verkürzt. Mit der Neuerstellung werden die Datenbank neu eingerichtet und die vorhandenen Dateien vom Magnetband übernommen.

Dialog

Das Abarbeiten des Dialogsystems erfolgt an den einzelnen Bildschirmen. Die Datenübertragung zum bzw. vom Bildschirm erfolgt über ein spezielles Assemblerprogramm. Dieses ermöglicht die bildschirmweise Arbeit (full-screen-prinzip).

Start des Dialogsystems

Das Dialogsystem wird mit dem Aufruf einer TSO-Prozedur gestartet, welche den Nutzer zum Löschen des Bildschirmes und zum Umschalten des Bildschirmmodus (Drücken Tasten CLEAR und PA3) auffordert. Nach diesen Maßnahmen meldet sich das Dialogsteuerprogramm mit der Aufforderung zur Eingabe der logischen Bildschirmnummer und dem für diesen Bildschirm gültigen Codewort. Bei falschen Eingaben werden die Anfragen wiederholt. Der Bediener hat jedoch die Möglichkeit durch Eingabe von „!" das Dialogsteuerprogramm zu beenden. Nach ordnungsgemäßen Eingaben wird der Bildschirm gelöscht und es erscheint die Mitteilung
START DIALOGSYSTEM UAS
Nun können die gewünschten Auskünfte mit Eingabe der entsprechenden Kommandos und der zugehörigen Parameter abgefordert werden.

Kommandoabarbeitung

Die Namen der Kommandos einschließlich der Parameter werden, mit Komma getrennt, auf der ersten Bildschirmzeile linksbündig eingegeben. Die zulässige Länge der Kommandonamen ist mit sechs Zeichen begrenzt.
– Systemkommandos
Systemkommandos sind Bestandteil des Dialogprogramms und brauchen nicht mit einer Kommandobeschreibungstabelle in die Datenbank aufgenommen zu werden.
KE[,Kommandoname].
Dieses Kommando dient der Kommandoerläuterung. Ohne Angabe eines Parameters wird eine Übersicht aller im Dialog möglichen Kommandos gezeigt. Von den Nutzerkommandos wird dafür die erste Zeile des Kommandomenüs herangezogen, welche diesem Zweck angepaßt sein sollte. Reicht ein Bildschirm für die Ausgabe nicht aus, werden Fortsetzungsbilder erzeugt. Als einziger Parameter ist der Name eines Kommandos zulässig. Es wird dann das mittels Kommandobeschreibungstabelle eingespeicherte Menü des angegebenen Kommandos auf den Bildschirm ausgegeben.
LB[,bildnummer].
Mit diesem Kommando können erzeugte Fortsetzungsbilder des vorhergehenden Kommandos auf dem Bildschirm ausgegeben werden. Ohne Angabe der Bildnummer wird bei jedem weiteren Kommandoaufruf das jeweils nächste Bild aus der Bilddatei entnommen, sonst das Bild mit der angegebenen Bildnummer. Die Bildnummer wird bei allen Kommandos im Mitteilungsfeld auf der rechten Seite der ersten Bildschirmzeile ausgewiesen. Das erste Bild hat dabei die Bildnummer zwei. Das ermöglicht im Fehlerfall, den Bildschirm durch Lesen des Bildes mit der Nummer eins neu zu formatisieren.
ENDE.
Das Kommando benötigt keine Parameter und beendet den Dialog. Anschließend wird der Nutzer zum Umschalten des Bildschirmmodus (Drücken Taste PA3) aufgefordert. Danach kann er wieder im normalen TSO-Betrieb arbeiten.
– Nutzerkommandos
Beim Aufruf eines Nutzerkommandos werden die Anzahl und die Art der notwendigen Kommandoparameter der eingespeicherten Kommandobeschreibungstabelle entnommen. Existiert in der Datenbank unter dem angegebenen Namen kein Kommando, wird dies dem Bediener im Systemfeld mitgeteilt. Mit Hilfe der Dateibeschreibungstabelle, welche die entsprechenden Dateiangaben enthält, und der dateieigenen Suchdateien werden die gewünschten Datensätze ausgewählt. Die Aufbereitung der Daten erfolgt anhand der zugehörigen Bildbeschreibungstabelle.

Schlußbetrachtung

Die erste Anwendung des vorgestellten Datenbanksystems erfolgte für den Projektkomplex Lieferplan, der im Stapelbetrieb abgearbeitet wird. Nach der Aktualisierung werden die betreffenden sequentiellen Dateien in die Datenbank eingespeist. Die vorgesehenen Auskünfte können über die vorhandenen TSO-Terminals bereitgestellt werden. Nach vorliegenden Erfahrungen beträgt der Qualifizierungsaufwand für die Handhabung des Datenbanksystems eine Woche. Die Projektvorbereitung für einen konkreten Anwendungsfall – Festlegung des Bildaufbaus, Einspeichern der Tabellen und der sequentiellen Dateien – wird mit einer Woche eingeschätzt.

Nutzerbezogene Bewertung automatisierter Informationssysteme (Teil 1)

Dr. Manfred Feder
Kombinat VEB NARVA
„Rosa Luxemburg" (BGW)
Dr. Günter Klatt
Humboldt-Universität zu Berlin,
Sektion Wissenschaftstheorie
und -organisation

Mit dem massenhaften Einsatz moderner, arbeitsplatznaher Informationstechnologien werden bewährte Vorgehensweisen der ökonomischen Bewertung der Effektivität automatisierter Informationssysteme allein nicht mehr ausreichend sein. Die Nutzer in ihren Arbeitskollektiven und -bereichen müssen als Persönlichkeiten berücksichtigt werden; das schließt auch die Entwicklung „nutzerbezogener" Bewertungsmethoden ein, die in Einheit von ökonomischen und sozialen Effektivitätsbewertungen automatisierter Informationssysteme entwickelt werden müssen. Außerdem sollen die Nutzer die Möglichkeiten erhalten, selbst auf die Informationssysteme und die Arbeitsgestaltung gezielt Einfluß zu nehmen.
Der Einsatz moderner Informationstechnologien erfordert eine Einschätzung ihrer Möglichkeiten und Grenzen in Abhängigkeit von den konkreten Anwendungszielen und -bedingungen, um die verschiedenartigen, z. T. neuen Wirkungen in den Anwendungsbereichen effektiv beherrschen zu können. Damit aber ergeben sich für die Bewertung automatisierter Informationssysteme (AIS), die moderne Informationstechnologien nutzen, neue Perspektiven.
Während etwa bis zur Mitte der 70er Jahre die zentralisierten, arbeitsplatzfernen AIS vorrangig nach technologischen und ökonomischen Gesichtspunkten bewertet und soziale Aspekte nur am Rande behandelt wurden, haben sich in den letzten Jahren international Auffassungen durchgesetzt, die unter dem Eindruck des massenhaften Einsatzes von Büro-, Personal- und Arbeitsplatzcomputern eine stärkere Berücksichtigung der Nutzer als soziale Subjekte und der Arbeitskollektive bei der Anwendung dezentraler, arbeitsplatznaher Informationstechnologien und -systeme fordern. Demzufolge müssen soziale Aspekte einbezogen und ein deutlicher Übergang von der Aufgaben- zur Nutzerorientierung bei der Gestaltung, Bewertung und Anwendung automatisierter Informationssysteme vollzogen werden. Das heißt, die zu automatisierenden Aufgaben werden künftig im Kontext der Gesamtaufgaben und -bedingungen sowie der Arbeitsorganisation und -inhalte der die Informationstechnologien nutzenden bzw. durch ihre Nutzung betroffenen Werktätigen zu sehen sein.
Auf der Grundlage eines zwischen dem VEB Kombinat NARVA und der Sektion Wissenschaftstheorie und -organisation der Humboldt-Universität zu Berlin abgeschlossenen Leistungsvertrages wurden auf der Basis dieser Erkenntnis Prinzipien einer Methodik der nutzerbezogenen Bewertung von AIS erarbeitet, die neben der ökonomischen auch ihre soziale Bewertung erlauben, wie erfolgreiche Erprobungen bewiesen haben. Damit besteht die Möglichkeit, den Grad der sozialen Akzeptanz vor allem arbeitsplatznaher automatisierter Informationssysteme entscheidend zu erhöhen: Die Nutzer können aus der Sicht ih-

rer Anwendungsbereiche zielgerichtet in die Bewertung des Prozesses und der Ergebnisse der Informationssystemgestaltung und -anwendung von Anfang an wirkungsvoll einbezogen werden. Mit der Erhöhung der sozialen Akzeptanz auf Grund von Verbesserungen der Bedingungen, der Organisation und des Inhalts der Arbeitstätigkeit wird zugleich eine Steigerung des ökonomischen Ergebnisses erwartet.

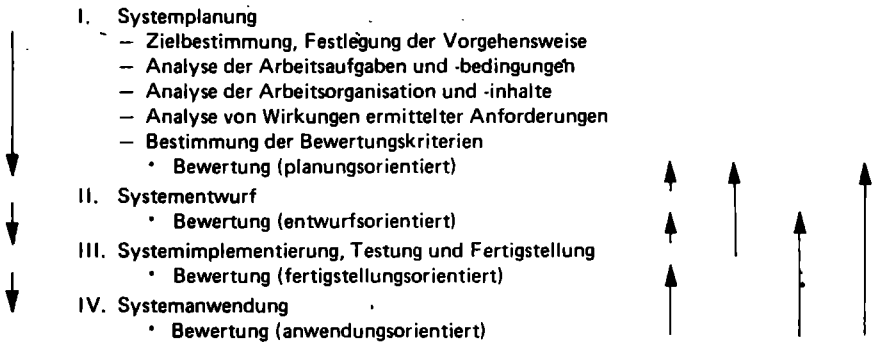


Abb. 1 Zyklus der nutzerbezogenen Bewertung von AIS

Zur Methodik einer nutzerbezogenen Bewertung

Aus einer vertiefenden theoretischen Untersuchung von Wechselwirkungen in der sozialen Organisation (Anwendungsbereich) als Bezugsrahmen der Bewertung (der Nutzer als Subjekt und das automatisierte Informationssystem als Objekt der Bewertung) wurde ein Zyklus der nutzerbezogenen Bewertung entwickelt, der sowohl die Anwendung von Methoden der Informationssystem- und Arbeitsgestaltung als auch eine Bewertung erforderlich macht (Abb. 1).

Die Phase der Systemplanung beinhaltet die Zielbestimmung der Bewertung sowie davon abgeleitete analytische Arbeiten der zu verändernden Arbeitsaufgaben und -bedingungen, Arbeitsorganisation und -inhalte. Daraus lassen sich Schlußfolgerungen über erste Anforderungen an die Gestaltung der Bewertungskomponenten soziale Organisation, Nutzer und AIS ableiten. Sie umfassen solche Anforderungen wie

– Neufestlegen der Verantwortung und Aufgaben
– Gestaltung der entsprechenden Arbeitsbedingungen (gemeinsam mit den Nutzern)
– möglichst selbständige Organisation der Aufgabenerfüllung unter Berücksichtigung der Erfordernisse der Arbeitsprozesse.

Die Ergebnisse dieser Analysen bilden die Grundlage von Wirkungsanalysen und Ableitungen zur Festlegung der Bewertungskriterien (Bewertungskriterien werden aus dem Anwendungsziel des zu beurteilenden automatisierenden Informationssystems abgeleitet). Zu den sozialen Bewertungskriterien gehören insbesondere Verantwortungsbewußtsein, Leistungsmotivation und -bereitschaft, Kreativität, Entscheidungsspielraum, Kollektiventwicklung und -festigung. Diese Bewertungskriterien bedürfen der weiteren Untersetzung mit beschreibenden Merkmalen und Aussagen, Werten etc. Sie werden in einem Bewertungssystem zusammengefaßt, das eine entscheidende Voraussetzung der folgenden ersten Bewertung ist, die sich an den Zielen der Planung orientiert, d. h. Festlegung von Anforderungen an die Qualität eines automatisierten Informationssystems und seine Integration zur Durchsetzung sozialer und ökonomischer Zielstellungen der sozialen Organisation, bevor mit dem Systementwurf begonnen wird. Damit ist eine Entscheidungsgrundlage für die nachfolgenden Arbeiten am Systementwurf gegeben.

Die Phase des Systementwurfs umfaßt (in Realisierung der Zielstellung und der damit einhergehenden Sollkonzeption) formalisierte und nichtformalisierte Systemspezifikationen, die zu bewerten sind (Unter formalisierten Systemspezifikationen werden Algorithmen, Datenstrukturen u. a. verstanden. Nichtformalisierte Spezifikationen umfassen Beschreibungen des organisatorischen Ablaufs, die Art und Weise der Kommunikation zwischen den Nutzern).

Zur Phase der Systemimplementierung, des Testens und der Fertigstellung gehören organisatorische Anpassungsarbeiten und der Probebetrieb mit der Bewertung erster, tatsächlich erzielter Effekte.

Mit dem Bewertungszyklus werden folgende Aspekte berücksichtigt:

- Entsprechend dem Ablauf (von der Systemplanung bis zur Systemanwendung) werden Bewertungen im Ergebnis der Bearbeitung jeder Phase vorgenommen. Sie sind zugleich Voraussetzung für eine Entscheidung der Arbeiten an den folgenden Phasen
- der Zyklus bedingt ein phasenspezifisches Vorgehen der Bewertung, das von der Art der vorliegenden Ergebnisse abhängig gemacht werden muß
- Bewertungsergebnisse der einzelnen Phasen müssen – obwohl sie phasenspezifisch ermittelt werden – miteinander vergleichbar sein, damit die Nutzer gezielten Einfluß auf den Gesamtprozeß der Bewertung nehmen können.
- die erste Phase der Bewertung verpflichtet die Nutzer zu einer Vielzahl analytischer Tätigkeiten, die weit über die bisherigen Erfahrungen der Einsatzvorbereitung hinausgehen. Hier wird die Entscheidung über den Verlauf der Systemgestaltung und -anwendung gefällt.

Die erste Phase ist somit die wichtigste im Bewertungszyklus; die Mitwirkung der Nutzer ist hier am umfangreichsten und vielfältigsten.

Opermann /1, S. 1085/verwendet für ein methodisches Vorgehen in diesem Sinne den Begriff der „Varietät", d. h. die Erzeugung von Entwicklungs- und Entscheidungsalternativen und deren Bewertung. Das Prinzip der ständigen Rückkopplung und eventuellen Modifizierung vertreten auch Floyd und Keil /2/ in ihrem Versionenansatz, der für jede Teilkomponente eines Gesamtsystems gilt.

Dieses methodische Vorgehen für die nutzerbezogenen Bewertungen wirft die Frage nach den Unterschieden und Gemeinsamkeiten mit bisherigen Methoden der Bewertung auf. Wesentlich ist die Feststellung, daß gegenwärtige Bewertungen zumeist allein von einer auf der aufgabenabhängigen Informationsbedarfsanalyse aufbauenden ökonomischen Nutzeffektsanalyse ausgehen /3, 4, 5, 6, 7/. Demgegenüber bezieht sich eine nutzerbezogene Bewertung auf Analysen, die sich an den Arbeitstätigkeiten, den Arbeitsaufgaben und Ausführungsbedingungen sowie Leistungsanforderungen orientieren /8/. Das führt zu einer tätigkeitsgebundenen Informationsanalyse, die die zu erreichenden sozialen und ökonomischen Nutzeffekte sowie die sich daraus ergebenden technologischen Anforderungen an die Gestaltung und Integration eines AIS in die Arbeitstätigkeit der Nutzer einer sozialen Organisation zum Inhalt haben muß /9/. Im Zusammenhang mit einer nutzerbezogenen Bewertung sind Informationsbedarfsanalysen als gestufte, sich an den Arbeitstätigkeiten orientierende Analysen durchzuführen (Abb. 2). Sie umfassen damit neben dem Informationsbedarf zur Lösung von Nutzeraufgaben auch Informationen über die jeweiligen Arbeitsbedingungen, die Arbeitsorganisation und -inhalte sowie über daraus entstehende Wirkungen auf die Bewertungskomponenten.

Anforderungen an die Überführung in die Praxis

Die Methodik der nutzerbezogenen Bewertung bedingt ein stufenweises Vorgehen mit innerbetrieblichen Anpassungsarbeiten bei ihrer gleichzeitigen weiteren Präzisierung. Im Rahmen einer Konzeption zur Einführung der veränderten Bewertungsmethodik im VEB Kombinat NARVA ist als erste Stufe das Ermitteln des sozialen Nutzens aus der Einführung automatisierter Informationssysteme vorgesehen, ohne daß der Zusammenhang zur ökonomischen Nutzeffektsrechnung hergestellt wird. Ist ein gesichertes Niveau der Ermittlung ökonomischer Nutzeffekte gegeben, kann man sich ausschließlich auf die fehlende soziale Komponente konzentrieren. Dabei ist es erforderlich, die soziale Nutzeffektsermittlung auf die Gesamtarbeitstätigkeit der Nutzer zu beziehen, d. h. auf deren Arbeitsaufgaben und -bedingungen, Arbeitsorganisation und -inhalte. Hierzu werden die An-

forderungen (Soll-Zustand), deren Wirkungen überprüft sind, in Form von Kriterien und ihre Realisierung (Ist-Zustand) gegenübergestellt sowie die dabei sichtbaren Veränderungen von den Nutzern bewertet. Da das automatisierte Informationssystem zum Komplex der Arbeitsbedingungen gehört, werden hierzu entsprechende technologische Kriterien vorgegeben, die in unmittelbarem Zusammenhang zur Arbeitsorganisation und zu den Arbeitsinhalten stehen.

Die zweite Stufe sieht eine Zusammenführung der sozialen und der ökonomischen Nutzeffektermittlung vor. Außerdem müssen beide zu technologischen Leistungsparametern des AIS in Beziehung gesetzt werden. Mit der dritten Stufe soll der wechselseitige Zusammenhang zwischen den qualitativ unterschiedlichen Bewertungskriterien, ihren Merkmalen und Werten analysiert und beurteilt werden, so daß differenziert auf die Informationssystem- und Arbeitsgestaltung Einfluß genommen werden kann.

Die innerbetrieblichen Anpassungsarbeiten beziehen sich vor allem auf

– eine Veränderung der Methodik der Einsatzvorbereitung
– die ideologische Bereitschaft zur Unterstützung einer veränderten Vorgehensweise der Einsatzvorbereitung.

Veränderungen der Methodik der Einsatzvorbereitung umfassen insbesondere:

1. Die Entwicklung bzw. Umgestaltung von Formblättern zur

– Erfassung des sozialen Nutzens aus der Einführung eines automatisierten Informationssystems von jedem Nutzer im Anwendungsbereich
– Antragstellung an den Projektanten mit dem Ausweis von sozialen Kriterien, die Forderungen bezüglich veränderter Arbeitsorganisation und -inhalte umfassen (hierbei werden ausgewählte Kriterien in das Formblatt aufgenommen. Sie enthalten die mit den Nutzern abgestimmten und verdichteten Ergebnisse der individuellen Nutzensermittlung)
– Ausarbeitung präzisierter Funktionspläne (die vereinbarten Arbeitsaufgaben und Arbeitsbedingungen werden im Zusammenhang mit den abgestimmten Kriterien zur Arbeitsorganisation und zu den Arbeitsinhalten für jeden Nutzer fixiert)
– Planung und Abrechnung des sozialen Nutzens in Übereinstimmung mit den abgestimmten und verdichteten Ergebnissen der Nutzensbewertung im Anwendungsbereich.

2. Die Änderung entsprechender Kombinationsordnungen

– zur Integration der sozialen Nutzensermittlung in die Nutzensrechnung überhaupt
– zum Festlegen des Stellenwerts der sozialen Nutzensermittlung durch Aufnahme ausgewählter Kriterien (hierfür kann der Projektierungsantrag gewählt werden) in den Erneuerungspaß und das Pflichtenheft.

Bei ökonomischen Nutzeffektsberechnungen ist es oft allgemeine Praxis, daß der verantwortliche Leiter in Abstimmung mit kompetenten Mitarbeitern Bewertungen vornimmt, so daß nicht unbedingt die Stellungnahme aller Nutzer abgefordert wird. Werden jetzt alle Nutzer in die Bewertung einbezogen und dabei die Einheit von sozialer und ökonomischer Nutzeffektsermittlung zur Disposition gestellt, ergeben sich neue, erweiterte Möglichkeiten der Mitwirkung am Arbeitsplatz.

Dabei zeigt sich, daß einerseits gewohnte Wege der Bewertung verlassen werden müssen und andererseits auf gesicherten positiven Erfahrungen aufzubauen ist. In diesem Spannungsfeld wird sich die nutzerbezogene Bewertung entwickeln, wodurch die schöpferische Mitarbeit aller Nutzer unumgänglich wird.

Von entscheidender Bedeutung für die Nutzermitwirkung ist die Transparenz der Methode. Sie ist eine Vorbedingung für die Mitwirkung des Nutzers. Er muß sich mit den Zielen und der Vorgehensweise, den verwendeten Begriffen und Zusammenhängen identifizieren können. Die Bereitschaft zur schöpferischen Mitarbeit wächst in dem Maße wie es gelingt, positive Veränderungen im Ergebnis von Bewertungen eines einzuführenden automatisierten Informationssystems am Arbeitsplatz durchzusetzen.

Aufgabenorientierte Bewertung	Nutzerbezogene (tätigkeitsorientierte) Bewertung
– Aufgabenanalyse	– Aufgaben- und Bedingungenanalyse (einschließlich Informationsbedarfsanalyse)
– Informationsbedarfsanalyse	– Analyse der Arbeitsorganisation und -inhalte (einschließlich Informationsbedarfsanalyse)
	– Analyse der Wirkungen ermittelter Anforderungen an die Bewertungskomponenten (einschließlich Informationsbedarfsanalyse)
– ökonomische Nutzeffektsanalyse	– sozialökonomische Nutzeffektsanalyse

Abb. 2 Gegenüberstellung der aufgabenorientierten und nutzerbezogenen Bewertung

Charakteristik der Arbeitsaufgaben und -bedingungen[1]
(im Zusammenhang mit vereinbarten Kriterien der Arbeitsorganisation und -inhalte[2]

Arbeitsaufgaben und -bedingungen	zu Kriterien der Arbeitsorgansiation und -inhalte

[1] Die Charakteristik umfaßt neben einer Kurzbeschreibung der Arbeitsaufgaben und -bedingungen die ihnen jeweils zuordenbaren Kriterien der Arbeitsorganisation und -inhalte als Ausdruck des zu erwartenden sozialen Nutzens für den Werktätigen.

[2] Kriterien der Arbeitsorganisation und -inhalte sind:
– Qualifikationsübereinstimmung
– Leistungsbereitschaft
– Schöpfertum/Neuererwesen
– Flexibilität/Mobilität
– Entscheidungsspielraum
– kollektive Zusammenarbeit

Abb. 3 Funktionsplan mit sozialen Kriterien

Literatur

/1/ Oppermann, R.: Aktive Akzeptanzunterstützung durch Betroffenenbeteiligung. – In: Office Management 32 (1984) 11. – S. 1084–1086
/2/ Floyd, C.; Keil, R.: Softwaretechnik und Betroffenenbeteiligung. – In: Mambrey, P.; Oppermann, R. – Beteiligung von Betroffenen bei der Entwicklung von Informationssystemen. – Frankfurt, New York 1983
/3/ Koreimann, D. S.; Methoden der Informationsbedarfsanalyse. – Walter de Gruyter. – Berlin, New York 1976
/4/ Parisini, E.; Wächter, O.: Organisationshandbuch für die Einführung von ADV-Systemen. – Berlin de Gruyter 1974
/5/ Schneider, H.; Köhli, S.; Systematische Projektierung der Datenverarbeitung. – Verlag Die Wirtschaft. – Berlin 1980
/6/ Wedekind, H.: Systemanalyse. – München 1976
/7/ Aktualisierung und Ergänzung zur Rahmenmethodik der Datenverarbeitungsprojektierung. – VEB Kombinat Robotron. – Berlin 1976
/8/ Klatt, G.: Theoretische und methodische Begründung einer nutzerbezogenen Bewertung automatisierter Informationssysteme (dargestellt am Beispiel des VEB Kombinat NARVA „Rosa Luxemburg" Berliner Glühlampenwerk). – Diss. B. – Humboldt-Universität zu Berlin 1987
/9/ Anordnung über Grundsätze für das einheitliche Herangehen an die Ermittlung, Planung und Nachweisführung des Nutzens und der Effektivität der Maßnahmen des wissenschaftlich-technischen Fortschritts. – Nutzensordnung. – In: GBl. Teil 1 Nr. 1. – Berlin 13. 1. 1987

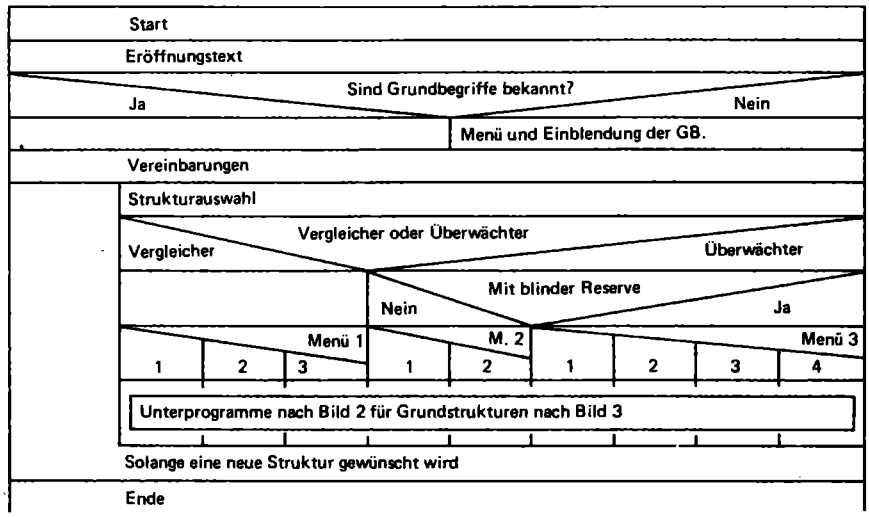
Dialogorientiertes Demonstrationsprogramm zur Wissensvermittlung

Jan-Markus Trauzettel
Ingenieurhochschule Mittweida

Der ständig steigende Umfang an zu vermittelndem Wissen verlangt in der Ausbildung der Studenten neue Methoden zur effektiveren Gestaltung der Lehrveranstaltungen. Neue Möglichkeiten bieten sich dabei mit der Einführung von Kleinrechnern (Z9001) zur Demonstration verschiedenartigster Prozesse und Probleme. Das vorgestellte Programm soll die Möglichkeiten der Dialogarbeit im Lernprozeß zeigen.

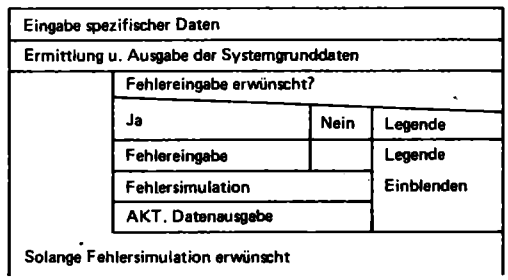
Problemstellung

Im vorliegenden Programm geht es darum, die Eigenschaften fehlertoleranter Systeme (deren praktische Realisierung Rechnersysteme, komplette Steuerungssysteme oder bei entsprechender Untersuchung in der Zukunft auch Softwaresysteme darstellen könnten) zu demonstrieren. Dabei soll der Nutzer einen Überblick zu den wichtigsten Leistungsparamtern solcher Systeme erhalten und einige mögliche Systemkonfigurationen kennenlernen. Um das Programm auch Anwendern ohne Vorkenntnisse zugänglich zu machen, werden die wichtigsten Grundbegriffe der Fehlertoleranz bereitgestellt. Auf die Erläuterung der Grundbegriffe muß an dieser Stelle verzichtet werden.



Ziel des Dialogs

- Mit dem Dialog Nutzer-Programm soll der Lernende angeregt werden, selbständig tiefer in die Problematik einzudringen, indem ihm Entscheidungsfragen bzw. Menüs zur Auswahl gestellt werden.
- Die natürliche Neugier des Menschen wird dazu ausgenutzt, um durch gezieltes Probieren alle möglichen Varianten von Problemlösungen und deren Vor- und Nachteile zu erkennen.
- Der Dialog bietet die Möglichkeit, den Nutzer in die an sich „vorprogrammierte" Lösung einzubeziehen und ihm durch die



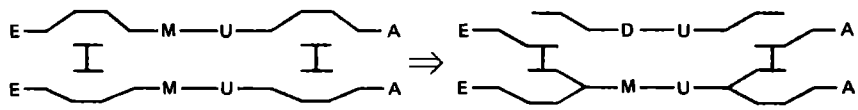


Abb. 1 Struktogramm des Hauptprogramms
Abb. 2 Struktogramm zum Unterprogramm – Systemuntersuchung
Abb. 3 Modifizierung der Grundstrukturen durch die Rekonfiguration am Beispiel der sich paarweise unterstützenden Einheiten

Grundbegriffe der Fehlertoleranz

Fehlertoleranz (nach Avizienis): Fehlertoleranz ist ein Überlebensattribut eines Systems, das darin besteht, einen Systemausfall auch dann zu vermeiden, wenn einzelne Fehler aufgetreten sind.

Statistische Redundanz: Es werden Ausgangssignale eines Systems miteinander verglichen und ein Mehrheitsentscheid herbeigeführt. Dadurch können $m < n/2$ Fehler toleriert werden.

Dynamische Redundanz: Sie ist gekennzeichnet durch das Vorhandensein von Reserveeinheiten. Ein Überwächter überprüft die gegenwärtig aktive Einheit ständig auf volle Funktionstüchtigkeit, im Fehlerfall wird eine sofortige Umschaltung auf die Reserveeinheit vorgenommen.

Blinde (leere) Redundanz: Bei der blinden dynamischen Redundanz führen die Reserveeinheiten im fehlerfreien Fall keine eigenen Aufgaben aus.

Funktionsbeteiligte Redundanz: Im Fall der funktionsbeteiligten dynamischen Redundanz führen die Reserveeinheiten im fehlerfreien Zustand eigene Aufgaben aus, die im Fehlerfall niedergelegt bzw. eingeschränkt werden.

Vergleicher: Baugruppe, die einen Mehrheitsentscheid zwischen den Ausgangssignalen eines Systems herbeiführt und eine Fehlermaskierung vornehmen kann.

Überwächter: Gerät, das die zugeordnete aktive Einheit auf volle, uneingeschränkte Funktionstüchtigkeit überprüft und im Fehlerfall eine Umschaltung auf die blinde oder funktionsbeteiligte Reserve vornimmt. Weiterhin wird von ihm ein Neuanlauf des Systems gestartet und eine Fehlerdiagnose an der fehlerhaften Einheit durchgeführt.

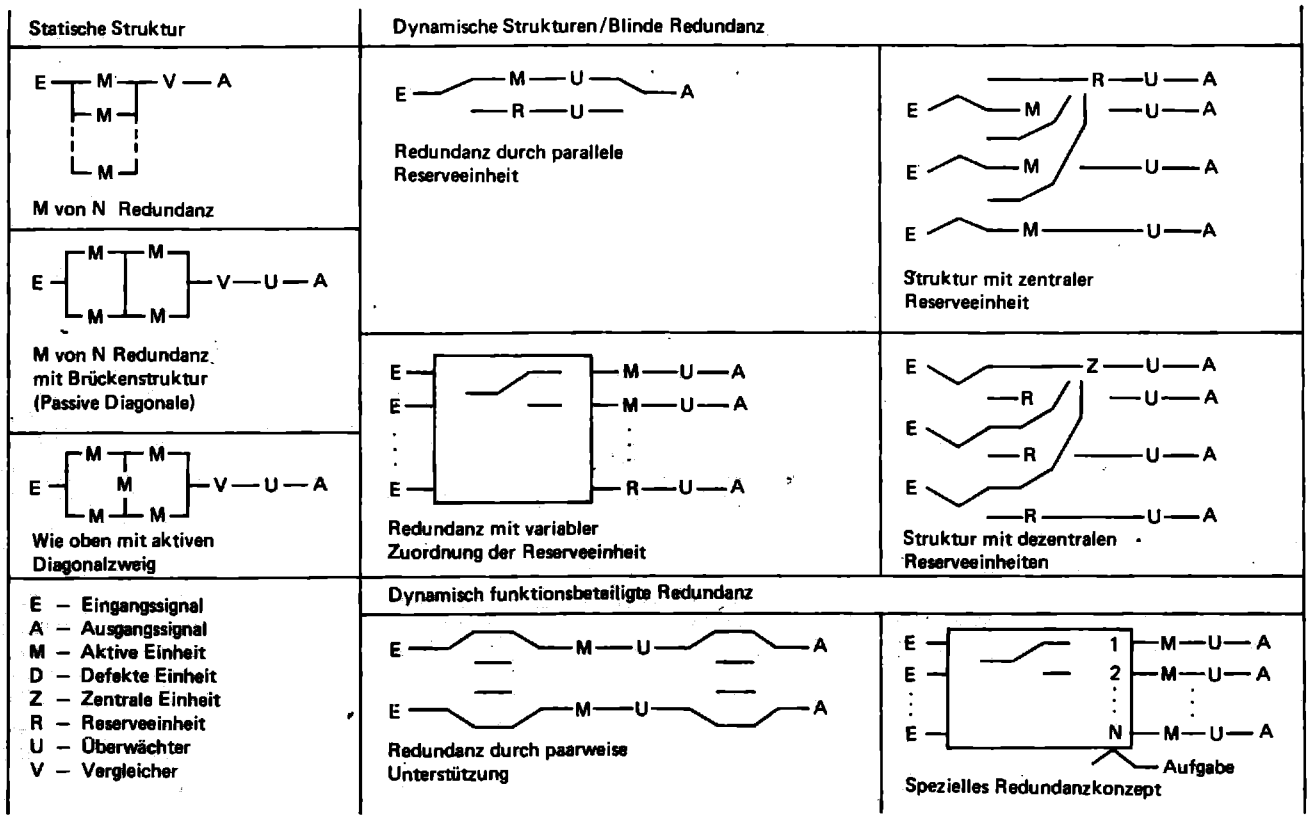


Abb. 4 Stilisierte Darstellungen der Grundstrukturen mit den Pseudografikzeichen des Z9001.

ständige Datenbereitstellung jede Auswirkung einer Information auf das System kenntlich zu machen.

- Unterstützung des Dialogs mit der Pseudografik der Kleinrechner (mit ihr ist es möglich, auf einfache Weise stilisierte Bilder darzustellen, die das Verständnis der Problematik unterstützen).

Kurzbeschreibung

Bei der Programmentwicklung wurde auf einen einfachen, leicht verständlichen Dialog Wert gelegt, um die Konzentration des Lernenden an die fachliche Problematik zu binden. Es werden nur Entscheidungsfragen oder Auswahlfragen gestellt, die von Menüs bzw. Bildern unterstützt werden. Abbildung 1 zeigt das Struktogramm des Hauptprogramms. Aus ihm sind die wesentlichsten Dialogstellen ersichtlich.

Zu Beginn des Programms werden dem Nutzer (auf Wunsch) die Grundbegriffe der Fehlertoleranz genannt und erläutert, so daß alle für das Verständnis des Programms notwendigen Begriffe der Fehlertoleranz als bekannt vorausgesetzt werden können. Weiterhin wird der Anwender mit den Randbedingungen (Vereinbarungen) bekannt gemacht die durch die Gegebenheiten des Rechners (Z9001) von Bedeutung sind. Danach kann mit der Strukturauswahl begonnen werden. Über einen Dialog wählt der Lernende eine Grundstruktur aus und kann diese nach eigenen Wünschen spezifizieren. Für diese vom Nutzer modifizierte Struktur ermittelt das Programm alle Hauptkennwerte der Fehlertoleranz (Abb. 2) und stellt sie mit der stilisierten Systemstruktur auf dem Bildschirm bereit. Danach beginnt das eigentliche Testen der Struktur, die der Lernende wiederum im Dialog mit dem Bildschirm vornimmt. Ihm werden die aktuellen Systemdaten und die modifizierte Struktur (Abb. 3) zur Verfügung gestellt bis diese nicht mehr arbeitsfähig ist.

Zur Demonstration der Funktionstüchtigkeit der in Abbildung 3 aufgeführten Strukturen wurde im Programm ein festes Testschrittmuster von 8 Bit ausgewählt. Die Einheiten werden als Negatoren (8 Bit) bzw. im Dialogzweig der Brücke als Identität simuliert. Bei der Programmerarbeitung wurde nur auf die theoretische Funktionstüchtigkeit Wert gelegt, technische Aspekte blieben unberücksichtigt. Die Zuverlässigkeit von Vergleicher und Überwächter wurden als Ideal vorausgesetzt. Die Fehlersimulation erfolgt, indem die jeweils eingegebenen Fehler in die Modulfunktion eingefügt werden und in der anschließenden Verarbeitung vom Vergleicher bzw. Überwächter erkannt werden. Die Erprobung des Programms zeigte, daß dialogorientierte Programme gegenüber rein demonstrierenden Programmen einen Lerneffekt bieten, weil der Lernende gezwungen ist, an der Problemlösung teilzunehmen. Programmseitig muß gesichert sein, daß eine eventuelle Fehleingabe ausgeschlossen ist, damit auch Nutzern ohne große Kenntnis des Kleinrechners eine einwandfreie Programmnutzung gesichert wird. Weiterhin zeigte sich der Einsatz der Pseudografik als sehr vorteilhaft und stellte sich als eine wesentliche Stütze des Lernprozesses dar, weil mit der bildlichen Darstellung von Grundstruktur und modifizierter Struktur eine bessere Vorstellung der Funktionsweise erzielt wurde.

Literatur

/1/ Bemmann H.-B.: Zuverlässigkeit und Fehlertoleranz in der Mikroelektronik Nachrichtentechn., Elektron. 25 (1985) 2, S. 75–77

/2/ Görke, W.: Was ist Fehlertoleranz? Elektronische Anlagen 26. Jahrg., 1984, H. 1 S. 29–31

Interaktive 3D-Modellierung

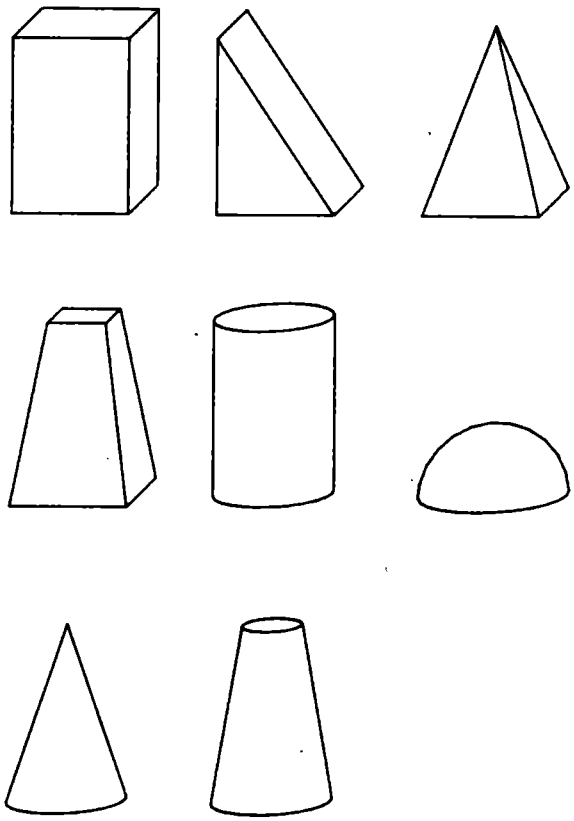
Roland Garz
Technische Universität Magdeburg
Sektion Informatik

Mit dem Ausbau der graphischen Datenverarbeitung tendiert die Entwicklung immer stärker zu 3D-Geometrieverarbeitungssystemen.

2D-Zeichnungserstellungssysteme sind bereits weit verbreitet und praxiswirksam. Bei 3D-Systemen gibt es einzelne Prototypen für bestimmte Anwendungen in Teilbereichen der Wirtschaft, aber einheitliche Richtlinien zur Entwicklung solcher Systeme liegen nicht vor.

Beispielsweise ist die Diskussion um die Definition der Modelle noch nicht abgeschlossen (CSG oder Boundaryrepresentation).

Dieser Beitrag stellt ein interaktives 3D-System – IGEMOS-3D –



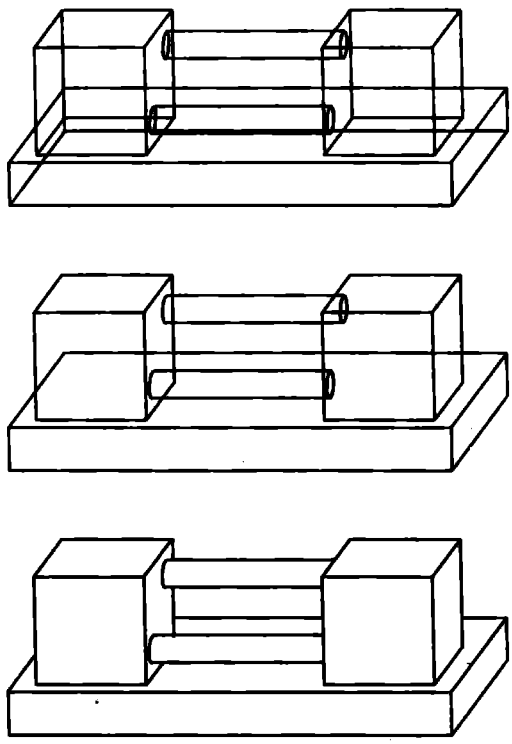


Abb. 1 Vorhandene Grundkörper in IGE-MOS-3D
Abb. 2 Modelliertes Gebilde und Hidden-Line-Algorithmus
a) Stufe 0,
b) Stufe 1,
c) Stufe 2

vor, das für die Modellierung im Maschinen- und Anlagenbau Verwendung findet.

Konstruktionsprinzipien

Um dreidimensionale Objekte darstellen und verarbeiten zu können, muß ein internes Modell zu ihrer Beschreibung vorhanden sein. Neben Drahtgitter- oder Flächenmodellen wird dazu hauptsächlich das Körper- oder Volumenmodell eingesetzt.
Für dieses System wurde das Volumenmodell gewählt, das auf der „Boundary representation" basiert.
Jeder Körper wird durch seine ihn berandenden Flächen beschrieben. Außer ebenen Flächen treten dabei einfach gekrümmte Flächen (Zylinder- und Kegelmantelflächen sowie deren Teile) und als doppelt gekrümmte Fläche die Kugelkappe auf. In diesem System findet keinerlei Zerlegung (Facettierung) der gekrümmten Flächen in ebene Flächen (z. B. Polyederflächen) statt. Alle gekrümmten Flächen werden in allen sie betreffenden Algorithmen (auch bei Hidden-line Algorithmen) als geschlossene Flächen behandelt. Daraus resultieren schnelle Algorithmen für die Darstellung von Objekten und für die Bearbeitung von verdeckten Kanten.
Den Nutzer berührt diese Randbeschreibung der Körper nicht, denn er befindet sich stets auf der Volumenebene, d. h. die kleinste Einheit, die er ansprechen kann, ist ein Volumen.
Die Modellierung seiner Objekte erfolgt auf der Basis der gedanklichen Zerlegung dieser Objekte in Grundkörper. Das bedeutet für den Nutzer Arbeit im interaktiven Dialog auf der Grundlage von Modellierungsalgorithmen mit Grundkörpern oder bereits aus Grundkörpern hervorgegangenen Körpern, die alle über vom Nutzer selbst vergebene Namen angesprochen werden.

Geometrieklassen

Für die geometrische Beschreibung von Objekten in den Speicherstrukturen RID (Rechnerinterne Darstellung) und EXD (Externe Darstellung) sind 5 Geometrieklassen erforderlich:
Klasse 1: Punkte
Elemente: Konturpunkte, Mittelpunkte, Hilfspunkte
Klasse 2: Kanten
Mit Kanten werden alle Elemente dieser Klasse bezeichnet, die eine echte konturbildende Funktion haben, unabhängig von ihrer geometrischen Form.
Elemente: Strecke, Kreis, Kreisbogen, Ellipse, Ellipsenbogen
Klasse 3: Flächen
Elemente: ebene Flächen, einfach gekrümmte Flächen, „doppelt gekrümmte Flächen.
Alle auftretenden Flächen müssen eine geschlossene Konturzugbegrenzung besitzen (die Kugelfläche z. B. erfüllt diese Bedingung nicht).
Ebene Flächen sind gerichtet, d. h. sie haben einen Flächennormalenvektor, der vom Material wegweist.
Klasse 4: Volumen
Ein Volumen ist eine räumlich geschlossene, zusammenhängende Punktmenge mit homogener Masseverteilung, die durch die Klassen 1 bis 3 beschrieben werden kann. Ein Volumen ist entweder ein Grundkörper oder ein Körper, der aus Grundkörpern durch Verknüpfungen oder Durchdringungen hervorgegangen ist.
Klasse 5: Bauteile
Ein Bauteil ist eine Zusammenfassung mehrerer räumlich getrennter Volumen (Klasse 4), die aus funktionaler Sicht eine Einheit bilden. Diese Volumen können nicht mehr verändert werden, auch nicht in ihrer räumlichen Lage zueinander. Bauteile können lediglich erweitert oder als Ganzes skaliert bzw. positioniert werden.

Rechnerinterne Darstellung eines Objektes – RID

Die RID bildet das Kernstück des interaktiven graphischen Systems und ist Grundlage für alle Manipulationen von Objekten, einschließlich ihrer Darstellung. Eine RID existiert für jedes Element der Klassen 4 oder 5 und besteht aus geometrieklassenspezifischen Speicherblöcken entsprechend der eingeführten Klassen 1 bis 5.
Die genutzte Programmiersprache PASCAL gestattet die Verwendung von dynamischen Variablen. Die RID eines Objektes ist eine Listenstruktur solcher dynamischer Variablen in Netzform. Die Elemente dieser Struktur sind jeweils die geometrieklassentypischen Speicherblöcke (POINTER im Sinne von PASCAL), die ihrerseits aus Adressverweisen auf Speicherblöcke der gleichen Klasse und auf Speicherblöcke tiefer gelegener

Klassen verweisen, soweit das möglich ist. Echte 3D-Weltkoordinaten befinden sich lediglich in der untersten Ebene (Punktspeicherblöcke).
Die RID befindet sich im Hauptspeicher und kann auf Grund ihrer dynamischen Struktur beliebig verändert werden.
Hinsichtlich des Datenzugriffes, der Wartung und Hauptspeicherauslastung ist eine solche dynamische Struktur bedeutend effektiver, als Strukturen, die mit der Programmiersprache FORTRAN erstellt wurden.
Die RID enthält für jedes geometrische Element des Objektes mindestens einen Speicherblock. Zusätzliche Verwaltungsblöcke existieren nicht. Das Gegenstück zur RID eines Objektes stellt eine geometrische Beschreibung außerhalb des Hauptspeichers, die EXD dar.

Externe Darstellung eines Objektes – EXD

Die EXD eines Objektes dient zur Archivierung der RID eines Objektes, die aus Grundkörpern durch verschiedene Modellierungsalgorithmen entstanden ist, für eine spätere Wiederverwendung. Diese Archivierung wird erforderlich bei Unterbrechung der interaktiven Arbeit mit späterer Fortsetzung oder zur Vermeidung von Wiederholarbeit beim Modellieren von Objekten, die mehrfach benötigt werden. Die EXD erfordert ein externes Speichermedium mit direktem Zugriff (Magnetplatte, Diskette ...). Alle Informationen der RID sind redundanzarm in stark komprimierter Form lückenlos in DA-Dateien abgespeichert. Das Auffinden einer EXD mit allen Informationen erfordert vom Nutzer lediglich die Angabe des EXD-Namens des Objektes. RID und EXD sind durch spezielle Algorithmen direkt ineinander überführbar.

Graphische Darstellung von Objekten

Grundlagen der graphischen Darstellung im System bilden die RID und GKS 2D. GKS ist eine standardisierte Grundsoftware für die Grafik, deren einheitliche Schnittstellen die Portabilität der Programme und die Unabhängigkeit vom jeweiligen grafischen Ausgabegerät (Vektorschirm oder Rasterschirme mit unterschiedlicher Auflösung) garantieren. Außerdem bietet das GKS ein standardisiertes Metafilekonzept und gestattet dem Nutzer die Verwendung eines Weltkoordinatensystems.
Als Projektionsart verwendet das interaktive System die schräge Parallelprojektion, wobei der Nutzer den Einfallswinkel der Projektionsstrahlen durch Angabe eines Standpunktes (Viewpoint) bezüglich des Nullpunktes des räumlichen Koordinatensystems bestimmen kann. Die senkrechte Parallelprojektion ist ein zugelassener Sonderfall.
Die RID des Objektes bleibt bei seiner Darstellung unverändert; es wird auch kein zusätzliches „graphisches Modell" erzeugt. Die Darstellung ebener Flächen erfolgt direkt als Projektion der in der RID enthaltenen Kanten. Die Darstellung gekrümmter Flächen wird aus der RID heraus generiert, denn an gekrümmten Flächen auftretende Sichtkanten sind abhängig von der Lage der Fläche und des Projektionsvektors und deshalb nicht Bestandteil der RID.

Modellierungsalgorithmen

Grundvoraussetzung für die Modellierung komplizierter 3D-Objekte ist die Generierung von Grundkörpern. Das System bietet bisher folgende Grundkörper zur Auswahl an (Abb. 1) – Aufnahme weiterer Grundkörper ist möglich:
– ebenflächig begrenzte Grundkörper
Quader, Keil, Pyramide, Pyramidenstumpf
– Grundkörper mit gekrümmten Flächen
Zylinder, Kegel, Kegelstumpf, Kugelkappe.
Nach Auswahl des Grundkörpers werden vom Nutzer die körpertypischen Dimensionen abgefordert und die RID wird generiert.
Eine zweite Gruppe der Modellierungsalgorithmen stellen Positionierungen von Objekten (Translation, Rotation) und Dimensionsänderungen (Skalierung) dar.
Die Verknüpfungs- bzw. Durchdringungsalgorithmen bilden die Gruppe von Algorithmen, die aus der Klasse der Grundkörper herausführen. Verknüpfung bedeutet, das Aneinanderfügen von Objekten an ausgewählten Kontaktflächen. Mit einer einzelnen Verknüpfungsoperation können jeweils zwei Körper an jeweils einer Kontaktfläche „zusammengeklebt" werden (Abb. 2a). Es werden zwei Arten von Kontaktflächen unterschieden: ebene und gekrümmte Kontaktflächen (KF). Es werden alle drei möglichen Lagen der Kontaktflächen zueinander behandelt: identische KF, sich überlappende KF und der Fall, daß eine KF vollständig in der zweiten enthalten ist.
Bei gekrümmten KF sind Verknüpfungen nur möglich, wenn die Krümmungen gleich sind, also bei „passenden" Komplementflächen.
Aus der Gruppe der möglichen Durchdringungen werden nur einige spezielle implementiert, nämlich solche, die besonders typisch für technische Gebilde im Maschinen- und Anlagenbau sind: ebenflächige Körper/ebenflächige Körper, ebenflächige Körper/Zylinder, Kegel, Zylinder/Zylinder (bei senkrecht aufeinanderstehenden Achsen).

Hidden-line-Algorithmen

Bei der Darstellung von komplizierten Objekten entstehen bei der Abbildung aller vorhandenen Kanten oft sehr unübersichtliche und teilweise sogar entstellende Bilder. Ein Mittel zur besseren Visualisierung ist das Entfernen der Kanten des Objektes aus dem Bild, die bei nicht transparenten Körpern unsichtbar wären. Der dafür angewendete Hidden-line-Algorithmus ist 3stufig, wobei eine höhere Stufe die unteren bedingt.
Die erste Stufe ist ausreichend für alle Grundkörper als Teilvolumen von Volumina und sonstige konvexe Objekte und arbeitet direkt im 3D auf der Grundlage der RID (Abb. 2b).
Die zweite Stufe ist erforderlich für direkt miteinander verknüpfte Teilvolumen und arbeitet, wie auch die dritte Stufe, auf der Grundlage der RID und im Bildraum des Objektes (Abb. 2c).
Die dritte Stufe wird angewendet, wenn mehrere räumlich voneinander getrennte Objekte bzw. Teilvolumen eines Objektes dargestellt werden sollen, die sich teilweise überdecken.
Die RID des Objektes als Grundlage des Algorithmus bleibt inhaltlich unverändert, allerdings wird sie bezüglich zu entfernender, zu verkürzender oder zu unterteilender Kanten erweitert. Diese Erweiterungen bleiben bei Translationen oder Skalierungen erhalten, werden aber bei jeder Rotation, jeder Änderung des Projektionsvektors und bei allen Modellierungsalgorithmen (Verknüpfungen, Durchdringungen) sofort rückgängig gemacht, da sich das Sichtbarkeitsverhalten verändert. Alle auftretenden gekrümmten Flächen, deren Behandlung ein Hauptproblem der Hidden-line-Algorithmen darstellt, werden durch einen jeweils speziellen Sichtbarkeitsalgorithmus behandelt und nicht durch ebene Flächen approximiert.

Nutzerschnittstellen des Modellierungssystems

Die erste Schnittstelle stellt das GKS mit seinem Metafilekonzept zur Verfügung. Es ermöglicht die Langzeitarchivierung von Bildern der modellierten Objekte in einer standardisierten Bilddatei, die portabel zu anderen mit GKS arbeitenden Systemen ist.
Als Ergänzung zum Metafilekonzept wurden zwei Möglichkeiten der graphischen Ausgabe von Bildern auf externe Geräte implementiert:
– Ausgabe auf DIGIGRAF 1612 (offline über Magnetband)
– Ausgabe über Seriendrucker SD 1154 (GRAF).
Eine weitere Schnittstelle liefert das System mit Hilfe der RID jedes Objektes. Es ist in jedem Modellierungsstadium möglich, die konkreten 3D-Welt-Koordinaten für anschließende, weiterführende Berechnungen auszugeben.
Der Anschluß eines Zeichnungssystems mit Bemaßung ist denkbar.

PSU und CCS für DOS-4/ES

Frank Gebhardt
VEB
Datenverarbeitungszentrum
Leipzig

Im VEB Datenverarbeitungszentrum Leipzig erfolgt im Fachbereich Forschung und Projektentwicklung (Abteilung F4) die Entwicklung und Bereitstellung der Programmsysteme PSU (Programmierung stützende Umgebung) und CCS (Computer Communication Software) für das ESER-Betriebssystem DOS-4/ES. Die Vorbildlösungen der PSU und CCS wurden vom VEB Leitzentrum für Anwendungsforschung für das Betriebssystem OS/ES entwickelt.

PSU 3.3 Modifikation 5

Die PSU bietet den Softwareprojektanten eine ideale Entwicklungsumgebung. Es werden Werkzeuge zur Erstellung, Pflege und Archivierung von Software bereitgestellt. Die PSU ist die UNIX-kompatible Soft-

ware im Betriebssystem DOS-4/ES. UNIX-kompatible Systeme werden auf nahezu allen Rechnertypen der DDR angeboten.
Der Vorteil für den Projektanten liegt in der gleichartigen Anwendung und Bedienung der UNIX-Systeme auf unterschiedlichen Rechnerarchitekturen. Zum anderen sind die meisten der Werkzeuge dieser Systeme portabel.
bar und austauschbar.

Leistungen der PSU 3.35

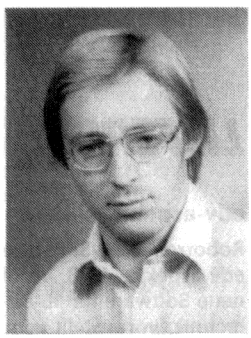
Die PSU arbeitet mit eigenen Treibern für Ein/Ausgabegeräte. Sie bietet dem Nutzer effektive reorganisationsfreie Filesysteme als Datenraum auf Magnetplatten. Es wird eine Basismenge von Kommandos bereitgestellt, die in jedem UNIX-System verfügbar ist. Für den Nutzer gibt es keine Unterschiede zwischen Datei-Ein/Ausgabe und Geräte-Ein/Ausgabe.
Der bereitgestellte C-Compiler bietet die Laufzeitumgebungen für PSU, OS und DOS-4.
Leistungsmerkmale:
● Hierarchische Dateiverwaltung
– baumstrukturierte Filesysteme
– integrierte Katalogverwaltung
● umfangreiches Kommandosystem
– etwa 150 Kommandos für Softwareentwicklung
– Kommandointerpreter SHELL
● einheitliche Ein/Ausgabe-Mechanismen
– E/A Umlenkungen
– Pipelines zwischen Programmen
● leistungsfähiger C-Compiler
– wahlweise Laufzeitumgebung PSU, OS-DOS-4
– SCREEN Programmierung für EC 7920.

Möglichkeiten der PSU 3.35

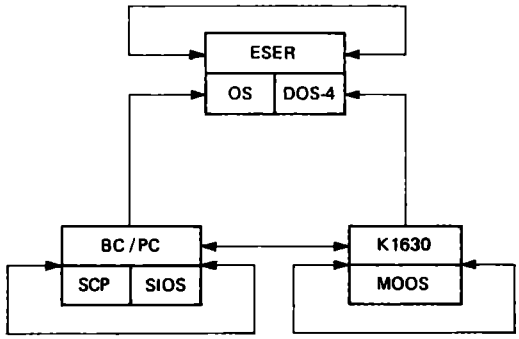
Der Nutzer kann mit der PSU die Erfassung, Bearbeitung und Wartung komplexer Programmsysteme effektiv durchführen. Die PSU bietet einen integrierten Ressourcenschutz für den exklusiven und geteilten Zugriff auf Filesysteme. Einzelne Kommandos lassen sich über Pipelines zu leistungsfähigen Werkzeugen verbinden.
Dem Nutzer werden PSU-Schnittstellen für Assembler und C zur eigenen Kommandoprogrammierung geboten:
– Speicherung, Verwaltung und Archivierung von Daten
– Mehrnutzerzugriff zu den gespeicherten Daten auf MP
– flexibler Datenschutz
– Bearbeitung der Daten mit leistungsfähigen Werkzeugen
– offene PSU-Schnittstellen
– Portabilität von Programmen, Kommandos und Werkzeugen.

Werkzeuge der PSU 3.35

In der PSU 3.35 werden umfangreiche Werkzeuge für die Softwareentwicklung angeboten. Der Vorteil ist auch hier, daß die meisten der Werkzeuge auf allen UNIX-Systemen in ihrer Anwendung identisch sind. Die wichtigsten Werkzeuge sind:
– Editoren
ED – Zeileneditor für Textfiles
VE – Screeneditor für Textfiles
HED – Screeneditor für Hexadezimalfiles
SED – Eingabestrom-Editor
MED – Modell-Editor
BED – Zeileneditor für Hexadezimalfiles
– Textverarbeitung
NROFF – Textprozessor für Zeilendrucker
TBL – Tabellengenerierung in Texten
ME/MS – Makroprozessor für NROFF
NEQN – Präprozessor für mathematische Ausdrücke
– Dateiverbindung zum DOS-4
IMP – Einlagerung von Daten aus DOS-4-Dateien in ein PSU-Filesystem
EXP – Auslagerung von Daten aus einem PSU-Filesystem in DOS-4-Dateien
– Pflege von Filesystemen
BKUP – physische Wartung
MKFS – Anlegen und Einrichten



**Frank Gebhardt (33) studierte von 1974–1978 an der Handelshochschule Leipzig in der Fachrichtung Mathematik und Datenverarbeitung. Er ist seit 1978 im VEB Datenverarbeitungszentrum Leipzig im Bereich Forschung tätig.
Als Themenleiter ist er seit 1986 für die Softwarekomponenten PSU und CCS für das ESER-Betriebssystem DOS-4 verantwortlich.**



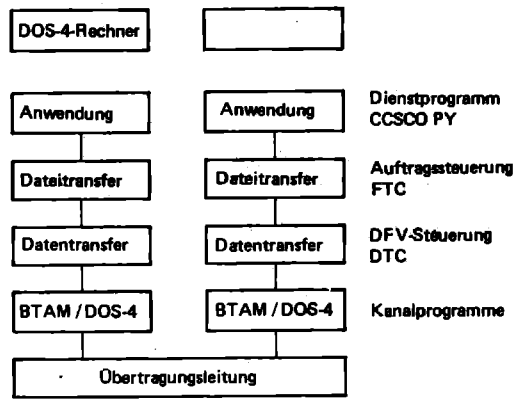


Abb. 1 Möglichkeiten den CCS für einen automatisierten Dateitransfer zwischen verschiedenen Rechnertypen unterschiedlicher Betriebssysteme
Abb. 2 Struktur des Dateitransfers zwischen zwei ESER-EDVA (FTC-File Transfer Control, DTC-Data Transmission Control)

DUMP – Entladen auf MB
RESTOR – Laden von MB
FSCK – Prüfen und Reparieren
DCHECK – Konsistenzprüfung
DEADBLK – Blöcke sperren
TAR – Bandarchivar
– Compiler des DOS-4
ASM – Assembler
COBOL – Cobol
FORT,FORTOC – Fortran
PL1,PI10C – PL1
RPG2,RPGA – RPGII mit AUTO-REPORT
PASCAL – Pascal
TMAC – Terminalmakro-Compiler
– Mustern von Textfiles
EGREP – Durchsuchen nach einem vollständigen regulären Ausdruck
FGREP – Durchsuchen nach einem festen Muster
GREP –Durchsuchen nach einem beschränkten regulären Ausdruck
– Drucken
PR,PRINT – Texte drucken
XD,OD – hexadezimaler, oktaler Druck
CHARS – Zeichensatz setzen
TABS – Tabulation setzen.

C-Compiler der PSU 3.35

Der C-Compiler bietet den Sprachumfang des „C Reference Manual" von D. M. Ritchie. Die Laufzeitumgebung der mit C entwickelten Programme kann wahlweise die PSU, das Betriebssystem OS/ES oder das DOS-4/ES sein. Die portablen Systemschnittstellen werden über die C-Bibliotheken realisiert:
– Standardbibliothek
– OS/ES Bibliothek
– DOS-4/ES Bibliothek
– Mathematische Bibliothek
– Quickscreen Bibliothek für EC 7920.
Zur effektiven Arbeit mit C werden unterstützende Werkzeuge angeboten:
CC – C-Compiler
CREFF – Cross-Referenz
CB – Strukturierung
LINT – Syntaxprüfung
QS – Präprozessoer für Quickscreen

M4 – Makroprozessor
YACC – syntaktischer Analysator
LEX – lexikalischer Analysator.

CCS 3.3 Modifikation 3

Die CCS bietet die Möglichkeit eines automatisierten Dateitransfers zwischen verschiedenen Rechnertypen mit unterschiedlichen Betriebssystemen (Abb. 1).
CCS-Entwickler sind:
OS – VEB LfA Berlin
DOS-4 – VEB DVZ Leipzig
SIOS – VEB LfA Berlin
SCP – VEB LfA Berlin
MOOS – VEB DVZ Berlin.
Die Rechnerkopplung erfolgt mit Mitteln der in der DDR derzeit verfügbaren Datendienste:
– Anschluß an das handvermittelte Datennetz (HDN) und/oder überlassene Standleitungen
– Multiplextsteuergerät für ESER-EDVA
– Modems und/oder Datennahübertragungseinrichtungen.

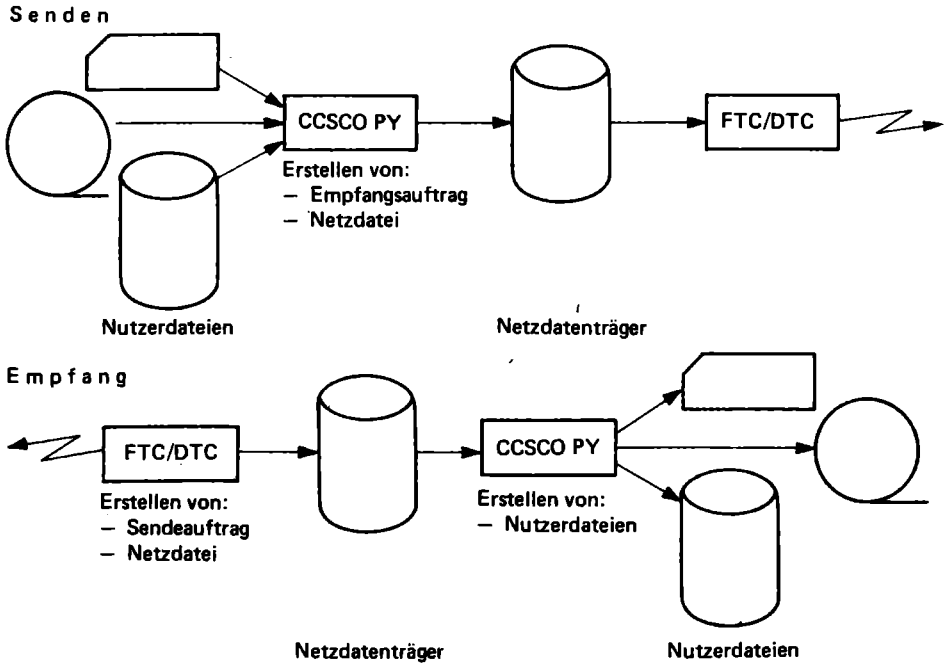


Abb. 3 Auftragssteuerung der CCS 3.33

Leistungen der CCS 3.33

Neben dem automatisierten Dateitransfer sind spezielle DFV-Dienste in der CCS 3.33 integriert:
– Leitungssteuerung mit BTAM (DOS-4/ES) synchron mit BSC1
– Fehlererkennungs- und -behandlungssystem
– Verwaltungssystem für Netztopologie
– Bedienersteuersystem
– Auftragssteuerung für Dateitransfer
– Prüfpunkt/Wiederanlauf
– Verwaltung von Verfalls- und Realisierungsfristen
– Dateitransfer über Transitknoten.

Dateitransfersystem der CCS 3.33

Das Dateitransfersystem ist programmtechnisch in Schichten eingeteilt, wobei die Prozesse der Datenbereitstellung und der Datenübertragung für den Nutzer getrennt sind. Die Abbildung 2 zeigt die Struktur beim Dateitransfer zwischen zwei ESER-EDVA mit dem Betriebssystem DOS-4/ES.

Auftragssteuerung der CCS 3.33

Die Auftragssteuerung wird mittels einer zentralen Auftragssteuerdatei realisiert. Die zu sendenden und die empfangenden Daten werden auf einem CCS-Netzdatenträger zentralisiert (Abb. 3).

Ablaufsteuerung der CCS 3.33

Das Übertragungssteuerprogramm wird ohne Generierung innerhalb eines Programmbereiches des DOS-4/ES (BATCH) aktiviert durch:
// ASSGN SYS100,
VOL=netzdatenträger
// EXEC CCS
Es realisiert die wichtigen Funktionen
– parallele Steuerung von maximal 5 Verbindungen
– automatische Arbeit von DTC und FTC nach Verbindungsstart
– Verarbeitung von asynchronen Steuerkommandos und
– Bedienerkommunikation mit Partnerrechnern.
Die wichtigsten Kommandos für das CCS-Programm sind:
USE – Verbindungsabbau
END – Verbindungsaufbau
MSG – Nachricht an Partnerrechner
RESTART – Wiederanlauf
INF – Informationen über die Verbindungen
DSP – Informationen über die Auftragssteuerdatei
SRT – Änderungen in der Auftragssteuerdatei.

Dienstprogramme der CCS 3.33

Das Programmsystem der CCS umfaßt eine Anzahl von Dienstprogrammen, die dem Nutzer umfangreiche Leistungen anbieten:

CCSCOPY – Zentrales Dienstprogramm der CCS-Anwendung
CCSPLAN – Druck der Auftragsdatei
CCSFORM – Anlegen der Auftragsdatei
CCSSORT – Sortieren der Auftragsdatei
CCSUNLO – Sicherung des Netzdatenträgers
CCSDISP – Übersicht über Auftragsdatei
CCSAUFT – Testhilfsprogramm.

Vertrieb von PSU 3.35 und CCS 3.33

Der Vertrieb der beiden Programmsysteme PSU und CCS für das Betriebssystem DOS-4/ES erfolgt durch den VEB DVZ Leipzig. Die PSU 3.35 wird bereits seit 1986 vertrieben.

Für die CCS 3.33 ist der Vertrieb ab Oktober 1988 vorgesehen. Im Rahmen der CCS-Entwicklung erfolgt zur Zeit die Implementatio der Software CCS/MV des VEB LfA Berlin für das Betriebssystem DOS-4/ES im VEB DVZ Leipzig.
CCS/MV dient zur Kopplung von verschiedener Rechentechnik auf Basis von asynchronen Verbindungen.

LIEFERBAR

edv-aspekte 4/88

Robotron-DCP-Software
edv-aspekte 4/88 veröffentlicht neue Software für 16-Bit-Rechentechnik, vorgestellt vom VEB Robotron-Projekt Dresden und VEB Robotron-Vertrieb Berlin:
Datenbanksoftware
– REDABAS-3
Unterschiede, Vorteile, Kompatibilität zum REDABAS/M16;
Dateien;
REDABAS-Programme;
– AIDOS/M-1 – Informationsrecherchesystem
Unterschiede, Vorteile, Kompatibilität zum AIDOS/M16;
Kompatibilität zum AIDOS/VS;
CAD/CAM-Software
– MultiCAD
– Grafisches Kersystem GKS16
– GRAMOS – grafische Basissoftware
Grafisches Modelliersystem;
Alphanumerische Datenhandhabung;
GRAMOS-Geometrie;
Integrierte Systeme
– ARIADNE
– MULTICOMP.
Leistungsstarke Hardware braucht leistungsstarke Basis- und Standardsoftware. Zur Qualifizierung und zum effektiven Einsatz Ihrer 16-Bit-Technik sollten Sie dieses Angebot nutzen.

Bestellungen für edv-aspekte 4/88 nimmt Ihr zuständiges Postamt, Abteilung Postzeitungsvertrieb, entgegen.

„Huckepack"-Betriebssystem

Das Betriebssystem CP/A der AdW der DDR für 8-Bit-Rechner eignet sich in Form eines normalen COM-Files, der als Kommando aufrufbar ist, für Disketten, deren Format von CP/A automatisch erkannt wird, als „Huckepack"-Betriebssystem:
Eine Diskette mit einem beliebigen Betriebssystem kann nach dem Aufruf der „Huckepack"-Datei unter CP/A arbeiten.
Der Vorteil der „Huckepack"-Methode ist, daß u. a. in einer SCP1715-, SCPX1526-, CP/Z- oder DAC-Installation ohne weitere organisatorische Neuregelungen die wesentlichen Vorteile und Möglichkeiten von CP/A unmittelbar nutzbar gemacht werden können.

Die von CP/A automatisch erkannten Diskettenformate sind aus der CP/A-Dokumentation (CPA.DOK) zu entnehmen.
In der hier zugrundeliegenden Installation der „Huckepack"-Systeme (CP/M, CP/L, CP/X, CP/Y) ist das DAC-Hausformat für PC 1715 (790 KByte, 2 Systemspuren, 192 DIR-Entries, 2 KByte Aufzeichnungsblockgröße, 5×1924 Byte/Spur) bei bespielten Systemspuren ("* * DAC") ebenfalls automatisch erkennbar. Nicht automatisch erkannt wird das Bürocomputer-SCPX-Format: 5×1024 Byte/Spur (OSBORNE-Format) bei 3 System-Spuren (DD-SS, 40 Spuren: 185 KByte).
Dieses Format wird mit dem DAC- bzw. CP/A-Hausformat DD-SS (40 Spuren: 190 KByte mit 2 Systemspuren) verwechselt.
Durch Voreinstellen des OFFSET auf 3 ist eine Verarbeitung zumindest lesend möglich. Falls auf dieser Art Disketten das Betriebssystem fehlt, werden sie als 200-KByte-CP/A-Datendisketten (OFFSET 0) erkannt.
Durch feste Formatvoreinstellung mit FORMAT[P] ist unter CP/A die Verarbeitung sehr vieler Diskettenformate möglich.
Die generierten „Huckepack"-Systeme sind als Maximalvarianten u. a. mit einem BIOS-Monitor versehen und damit vorrangig für Sonderanwendungen oder Schulungszwecke geeignet.
Interessenten wenden sich bitte an den VEB Weiterbildungsakademie des Kombinat Datenverarbeitung, Georgenstr. 23, Berlin, 1080

Dr. Hanisch

Abb. „Huckepack"-Systeme für PC 1715 und BC A 5120/30

PC 1715

CP/M.COM – Generiertes CP/A-Betriebssystem mit 2 Laufwerken vom Typ 1.6
CP/L.COM – Generiertes CP/A-Betriebssystem mit 2 Laufwerken vom Typ 1.2

FORMAT.COM	– Formatierungs-, Formateinstellungs- und physisches Kopier-Programm von CP/A
CPA1715G.COM	– Einrichten einer Systemdiskette von CP/A
SUBM.COM	– SUBMIT-Kommando für AUTOEXEC-Zwecke beim Kaltstart des „Huckepack"-Systems
AUTOEXEC.SUB	– SUBMIT-Datei für automatisches Anlaufen gewisser Aktionen beim Kaltstart – kann vom Nutzer beliebig gestaltet werden
CPA.DOK	– Betriebssystembeschreibung zu CP/A
CP/X.DOK	– Vorliegende Beschreibung

BC 5120/30

CP/X.COM – Generiertes CP/A-Betriebssystem mit 4 Laufwerken vom Typ 1.2
CP/Y.COM – Generiertes CP/A-Betriebssystem mit 3 Laufwerken vom Typ 1.6

FORMAT.COM	– Formatierungs-, Formateinstellungs- und physisches Kopier-Programm von CP/A
CPABCGEN.COM	– Einrichten einer Systemdiskette von CP/A z. B. aus dem „Huckepack"-System
SUBM.COM	– SUBMIT-Kommando für AUTOEXEC-Zwecke beim Kaltstart des „Huckepack"-Systems
AUTOEXEC.SUB	– SUBMIT-Datei für automatisches Anlaufen gewisser Aktionen beim Kaltstart – kann vom Nutzer beliebig gestaltet werden
RAMDISK.SUB	– SUBMIT-Datei zum Füllen der RAM-Disk auf dem Laufwerk M bei vorhandener EM256-Karte
CPA.DOK	– Betriebssystembeschreibung zu CP/A
CP/X.DOK	– Vorliegende Beschreibung

KC-Datenbanksystem RELA

Die Computer-Hardwareproduktion des VEB Mikroelektronik Mühlhausen gibt Anwendern der verschiedensten Bereiche der Volkswirtschaft und gesellschaftlichen Einrichtungen ein neuartiges Arbeitsmittel in die Hand. Mit dem Betriebssystem CAOS sind der KC 85/2 und seine Folgerechner für die wichtigsten Aufgaben, die ein Computer zu lösen hat, bestens vorbereitet. Der Nachteil dieses auf Kassettenarbeit orientierten Systems ist die fehlende CP/M – Kompatibilität einschließlich der dazu notwendigen Diskettenstation. Da die Nachrüstung dieses Standards relativ aufwendig ist und den Anwender vor völlig neue Probleme stellt, wurde als Alternative der Baustein Rechnungslegung / Wirtschaftsführung (RELA) als Hardwareerweiterung des KC 85/3 entwikkelt. Der Leistungsumfang soll hier kurz beschrieben werden.

Technische Voraussetzungen

Neben dem in den KC 85/3 integrierten RELA-Baustein benötigt der Anwender entsprechend des zu lösenden Problems:
– Monitor (Fernsehempfänger)
– Massenspeicher (Kassettenmagnetbandgerät)
(wahlweise Turbo-Recorder)
– Druckwerk (mit V.24/Centronics-Schnittstelle)
(S3004 – mit/ohne Interfacebox)
– RAM-Erweiterung (Tab. 2)
– Busdriver D002 (Tab. 2)
– M003/M001 (Schnittstellenmodul)
Schaltkreisbasis:
EPROM (8 K * 8 Bit) Typ 2764
RAM (256 K * 1 Bit) U61256

Anwendungsgebiet

Das Anlegen und programmierte Verknüpfen mehrerer, in sich geschlossener Dateien ermöglicht das Lösen anspruchsvoller Aufgaben mit dem KC.
Das RELA-Konzept ist für den Anwender *offen,* d. h. vorgegeben sind die Dateiorganisation und alle wichtigen Funktionen, die zum Erstellen und Verwalten eines Datenbestandes notwendig sind. Eingeschlossen sind dabei eine RAM-Floppy für Daten und Anwenderprogramme, ein automatisches System zum Laden und Retten der Dateien sowie ein Festkomma-Arithmetikmodul.
Im einfachsten Anwendungsfall werden Dateien erstellt und als Stichwortkartei genutzt. Bei der Auswahl der Funktionen, die der Nutzer zur Verfügung hat, standen die Belange privater Handwerks- und Dienstleistungsbetriebe im Vordergrund. Die Initialisierung spezieller Anwenderprogramme, wie Rechnungslegung, Inventur, Kassenbuch, Warenzugang und Bestellauslösung, erfolgt durch Interpretation der ersten Datensätze der Anwenderdatei. Somit kann der Anwender die für ihn entwickelte Variante verschiedenen Bedingungen anpassen. Das Einbinden von Nutzerroutinen in die Dateien und deren Aufruf werden vom System unterstützt.

Soft- und Firmware

Kernstück des RELA-Bausteins ist ein dem Anwender angepaßtes und als Firmware ausgelegtes Datenbankbetriebssystem, das von 0C000H-0DFFFH adressiert ist. Nach erfolgter Modifizierung kann der KC mittels eines Dreh- oder Kippschalters wahlweise als RELA- oder BASIC-Rechner genutzt werden.
Zusammen mit dem RELA-Baustein wird eine Kassette mit definierten Anwenderdateien geliefert. Der Kundendienst des ELECTRONIC-SERVICE umfaßt die Vermittlung der RELA-Interessenten an einen Software-Entwickler, der Anwenderprogramme nach Kundenwunsch erstellt und in die RAM-Floppy des RELA-Systems einbindet. Ein Katalog der verfügbaren Standardsoftware und der RELA-Nutzerprogramme soll Mehrfachentwicklungen verhindern.

Die wichtigsten RELA-Systemkomponenten

In Abb. 1 ist das Zusammenspiel der einzelnen Module des Datenbanksystems skizziert. Die Software gehört nicht zum Lieferumfang der Systemlösung. Sie kann aber, wie Abb. 2 in einem Beispiel zeigt, eine wesentliche Leistungserweiterung darstellen.

– Die konsequente Anwendung der Menü- und Fenstertechnik ersetzt eine spezielle Datenbanksprache. Die Standardfunktionen (Tab. 1) lassen sich auch für den Einsteiger sehr leicht in einem gesonderten Fenster durch Cursorsteuerung anwählen. Die Verkettung von mehreren Menüpunkten erweitert das Angebot für den Dialogbetrieb auf mehr als 50 Funktionen.

– Die Dateisteuerung verwaltet Speicherblöcke zu je 32 KByte RAM bei Verwendung von Modulen mit einer Kapazität von 64 KByte oder 256 KByte. Spezielle Laderoutinen (CAOS-Format, V.24-Turbo, EPROM-Modul) sind so mit der Dateisteuerung verbunden, daß alle benötigten Dateien in einem Zuge auf die RAM-Ebenen gelesen und von dort gerettet werden können.

Diese Dateiorganisation ist auch außerhalb von RELA aus dem CAOS-Betriebssystem nutzbar. Es kann bei Beachtung der Modulsteuerung in BASIC, FORTH, EDAS und TEXOR in mehreren Ebenen gearbeitet werden.

– Die Standardfunktionen (Tab. 1) werden im Dialog ausgeführt. Sind die Dateien vom Anwender mit zusätzlichen Steuerdaten versehen worden, wird ein RELA-Interpreter vor jedem Aufruf der RAM-Floppy aktiv. Dieser leitet die vordefinierten Programmabläufe ein oder modifiziert die aufgerufene Systemfunktion.

– Das unter RELA-KALK erreichbare Menü bildet automatisch die Ladebefehle der Anwender- und Dienstprogramme. Das Nachladen der Zusatzsoftware von RAM-Floppy, EPROM oder Kassette ist Grundvoraussetzung für alle programmgesteuerten Datenmanipulationen.

– Die Anwenderschnittstelle (USER-Interface) ist ein Teil der Nutzerdatei. Zu den schon beschriebenen Aufgaben kommen u. a. die Festlegung und Bezeichnung der Arithmetikregister sowie deren Bedeutung (Programmausschriften), Tabellenformate, Listenköpfe und Bedienhinweise. Auch Standards (Anzahl der Vor- und Nachkommastellen, Keyboardfunktionen, Funktionstastenbelegung) sind somit jederzeit dem Anwenderwunsch angepaßt.

– Die RELA-Datei ist, abgesehen von einigen Steuer- und Sonderzeichen, vollständig kompatibel zum Textverarbeitungssystem TEXOR/SORED des VEB Mikroelektronik Mühlhausen.

Um unbeabsichtigte Veränderungen der Datei zu vermeiden und unberechtigtem Eingriff in einzelne Datensätze vorzubeugen, besteht die Möglichkeit einer Passworteingabe. Gesperrt sind dann das Editieren und die Anwenderprogramme, in denen ein Schutz vorgesehen ist.

Das modulare Konzept des KC 85 kommt dem Anliegen von RELA sehr weit entgegen. Entsprechend der hardwaremäßigen Ausstattung des Anwenders kann RELA als Hilfs- und Dienstprogramm oder auch als Datenbankbetriebssystem für die verschiedensten Verwaltungsaufgaben dienen.

Für die Nutzung der elementaren Funktionen werden beim Anwender kaum Erfahrungen im Umgang mit dem Computer vorausgesetzt. Volle Auslastung des Befehlsvorrates durch Funktionsverkettung bedarf jedoch einer intensiven Beschäftigung mit System und Dokumentation.

Interessenten wenden sich bitte an den ELECTRONIC-SERVICE, Helmut Ulbricht, Göttiner Landstr. 7, Brandenburg 1800, Tel.: 522076

Nähere Auskünfte erteilt **Uwe Zierott**, Tel.: Lehnin 294

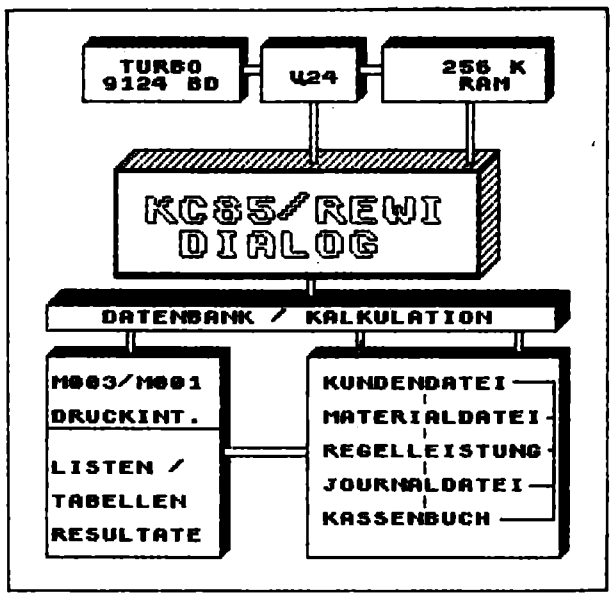
```
>STICHWORT<   *LADEN ---*   +REWI-KALK+   =RECHNUNG =   ^TEXTMODUS^
>SORTIEREN<   *RETTEN---*   +SOND.SUCH+   =KASS-BUCH=   ^-  n/216"^
>ANZEIGEN <   *<->KETTEN*   +<>DATEIEN+   =INVENTUR =   ^kbd<=>KBD^
>-  on/off<   *DATEI NEU*   +-  on/off+   =-  on/off=   ^-  on/off^
>EDITIEREN<   *HAUPTPROG*   +INIT-MENU+   =LAGERHALT=   ^-/TABELLE^
>ERSTELLEN<   *CAOS-MENU*   +SOND.PROG+   =RAMFLOPPY=   ^FESTKOMMA^
```

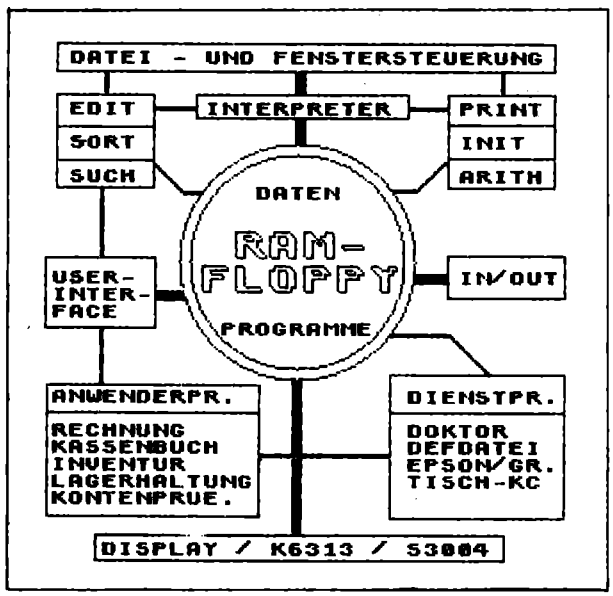
Tab. 1 Standardfunktionen – Menüangebot

MODULSCHACHT 08	0C	10	14	18	1C	KByte	Druck	Dateien
–	–					16	nein	1
M003	–					16	ja	1
M003	M022					32	ja	1
M003	M011					80	ja	3
M003	M256					272	ja	9
M011	M011					144	nein	5
M003	M011	M011	M011	M011	M011	336	ja	11
M003	M256	M256	M256	M256	M256	1272	ja	41

Tab. 2 Leistungsfähigkeit von RELA in Abhängigkeit von der Gerätekonfiguration (mit und ohne Busdriver)

Abb. 1 RELA-Systemkomponenten
Abb. 2 Anwendungsbeispiel





uniNet/DOS Version 1.2 uniNet/DCP Version 1.2

Softwarebezeichnung	uniNet/DOS uniNet/DCP	PASCAL/C
Nutzungsbeginn	1/89	9/88
Betriebssystem	DOS16, DCP ab Version 3.2 IBM-PC-DOS ab Version 2.1 MS-DOS ab Version 2.11	OS, DOS, SVM
Rechnertyp	16-Bit-PC, IBM-PC/XT IBM-PC/AT EC 1834, AC 7150	ESER
Ressourcenbedarf	256 KByte RAM 605 KByte Speicherplatz auf Diskette/ Festplatte	–
spezielle Peripheriebedingungen	Terminal-Leitung zu einem 32-Bit-Rechner, IBM-PC/XT: COM-Karte IBM-PC/AT: COM-Karte oder FPU-Karte EC 1834: ASK-Karte A 7150: ASP-Karte	Nutzung von PSU (alle ESER-Systeme)
Anwendungscharakteristik	Integration von 16-Bit-PC in 32-Bit-Rechner-Netze	Programmentwicklung, Portierung von PASCAL-Software
Anbieter	**VEB LfA Softwarebetrieb CAD/CAM, Jacques-Duclos-Str. 47–52, Berlin, 1156, Tel.: 37 80 33 60, Genth**	**Humboldt-Universität zu Berlin, Sekt. Mahtematik, PSF 1297, Berlin, 1086, Tel.: 20 93 23 48, Dr. Hohberg**

Kurzbeschreibung

Anwendungsbereich von uniNet Version 1.2

Mit der Einführung der neuen 32-Bit-Rechnergeneration des Kombinates Robotron wird mit SKRnet zugleich ein hochleistungsfähiges Rechnernetz angeboten, das hinsichtlich Schnittstellen und Protokollen kompatibel zu international weit verbreiteten Netzen ist. Die schrittweise Integration anderer Rechnerklassen und Rechnertypen in dieses Netz, vor allem der neuen 16-Bit-PC, gibt dem Anwender die Mittel für den Zugriff auf im Netz verfügbare Ressourcen unter Wahrung der lokalen Möglichkeiten der Informationserstellung, Informationsverarbeitung und Informationsverwaltung. Mit uniNet (**uni**versal **N**etwork **e**nhanced **t**ools) wird eine Software zur Verfügung gestellt, die es den Benutzern von PC mit DOS-kompatiblen Betriebssystemen gestattet, ihr arbeitsplatzbezogenes Gerät in das Netz der 32-Bit-Rechner zu intergrieren und deren Dienste direkt zu nutzen. Es sind darüber hinaus Funktionen zur bequemen Nutzung der gegenüber dem PC erheblich umfangreicheren Ressourcen der 32-Bit-Rechner enthalten.

Die im Netz zusammenarbeitenden Computersysteme (Netzknoten) können ihre Ressourcen geteilt nutzen sowie Informationen, Files und Programme austauschen. PC werden als Endknoten in das Netz eingebunden. Von einem solchen Endknoten aus sind alle Knoten des Netzes erreichbar, indem die Vermittlungsfunktion anderer Netzrechner (Vermittlungsknoten) genutzt wird.

Gegenüber uniNet Version 1.1 können nun auch Anwenderprogramme netzweit kommunizieren.

Leistungsumfang von uniNet Version 1.2

uniNet bietet folgende Netzdienste:

Netzverwaltung
– Steuerung und Überwachung der uniNet-Komponenten
– Konfigurierung des Netzes

Zugriff auf Files an entfernten Knoten
– Speichern und Rückgewinnen von Informationen an entfernten Knoten

Fileübertragung
– Austausch von Files mit entfernten Knoten

Geteilte Nutzung von Ressourcen
– gemeinsame Nutzung physischer Geräte (Drucker, Plattenspeicher) mit entfernten Knoten

Elektronische Post (MAIL)
– Senden von Nachrichten an andere Knoten des Netzes

Terminalemulation
– Terminalsitzungen an entfernten Knoten

Arbeit mit Server-PC
– Auf die Ressourcen der Server-PC kann netzweit zugegriffen werden

Programmkommunikation
– Netzweite Kommunikation von Anwenderprogrammen (PASCAL, C, FORTRAN, Assembler) unter Verwendung von Standardfunktionsaufrufen.

uniNet Version 1.2 realisiert den Anschluß von Personalcomputern an das Netz der 32-Bit-Rechentechnik auf Basis folgender Hardwarevoraussetzungen:
– asynchron V.24 (PC/XT, PC/AT, EC 1834 und A 7150)
– asynchron V.24 über FPU (PC/AT)
– IFSS (EC 1834 und A 7150).

Dabei wird eine vorhandene Terminalleitung zeitweilig oder permanent als Netzleitung definiert. Der Datenaustausch erfolgt nach einem netztypischen Protokoll.

Weiterentwicklung von uniNet

Weiterentwicklungsarbeiten von uniNet laufen in Richtung der Unterstützung von lokalen Rechnernetzen auf Basis ETHERNET-kompatibler Übertragungsgrundlage - (uniNet/DOS Version 2.0) und Bereitstellung eines Moduls für den Zugriff zu virtuellen Laufwerken lokal auf 32-Bit-Rechnern. Daneben werden Anpassungen an weitere Rechner und Betriebssysteme sowie die Unterstützung serieller synchroner Datenübertragung mit Modemsteuerung vorbereitet. Entsprechende Entwicklungstermine liegen in den Jahren 1989/90.

PASCAL/C

PASCAL/C erlaubt die Umsetzung von PASCAL-Programmen in C-Programme (Portabler PASCAL-Compiler – PCC).

Kurzbeschreibung

– Standard-PASCAL (Wirth) wird nach Standard-C (Kerninghan/Ritchie) übersetzt (inclusive wichtigste portable Sprachelemente von T-PASCAL). Auch bei der Übertragung von PASCAL-Dialekten liefert der Compiler brauchbare Resultate.
– Der 4-Pass-Compiler und das weitgehend portable Laufzeitsystem sind in C implementiert und wurden bereits auf P8000 (unter WEGA), IBM-PC-Kompatiblen, K 1630 (unter MUTOS) und ESER (unter ESER-Betriebssystem mit PSU) installiert; weitere Installationen erfolgen bei Bedarf.
– Für die Übersetzung eines PASCAL-Programms in die Maschinensprache sind die Schritte PASCAL → C, C → Maschinensprache erforderlich. Das Zeitverhältnis zwischen beiden liegt bei 1/5 und 1/10. Dazu existiert für jede Installation eine Kommandoprozedur.
– Großer Wert wurde auf die gute Stabilisierung der Compilation nach dem Auftreten von Quelltextfehlern gelegt.
– Eine zusätzliche Kommandoprozedur realisiert eine T-PASCAL-ähnliche Arbeitsweise beim Erkennen und Korrigieren im PASCAL-Quelltext.
– Die erzeugten C-Programme sind effektiv und gut lesbar.
– Einsatzbereiche des Compilers:
Portierung von PASCAL-Software in UNIX-kompatible Systeme Verbinden von PASCAL- und C-Quelltexten, selbständiger PASCAL-Compiler.
Als bisher größter Anwendungsfall wurde ein in PASCAL vorliegender PROLOG-Interpreter (reichlich 4000 Quelltextzeilen) mit Hilfe des PCC nach C transformiert und auf mehreren Rechnern in die Routinenutzung (Studentenpraktikum) überführt.

Verkaufen zum Zeitwert

450 8"-Disketten

(unbenutzt)

VEB Fernwasserversorgung
Elbaue – Ostharz Torgau
Abt. Rechentechnik
Naundorfer Str. 46
Torgau, 7290
Telefon: 5 70

Suchen dringend

Leporelloaufsatz für SD 1152

eventuell Tausch gegen

OEM-Baugruppen (Zeitwert).

VEB Institut für Getreideverarbeitung
Direktorat Projektierung/ Realisierung
Brunnenstraße 193/194
Berlin, 1054
Telefon: 2 80 51 61, App. 23

Bieten zum Zeitwert:

2 MC 80.21/1 m. EPROM-Programmiereinr. u. Druckersteuerung (parallel)

2 Kassettenlaufwerke K 5221

2 SD 1156 m. Parallelinterface (Auftisch- oder Standgerät)

1 SD 1154 m. Parallelinterface (Auftisch- oder Standgerät)

1 SD 1157.264 m. Seriellem Interface (Auftischgerät)

(alle Geräte neuwertig)

Suchen

EPSON-Drucker

(oder anderen kompatiblen Drucker)

Plotter (A 2-, A 3- oder A 4-Format)

auch Kauf/Verkauf

VEB Berlin-Chemie
Abt. ITC, Koll. Kittel,
Glienicker Weg 125, PF 9
Berlin, 1199
Telefon: 6 79 33 88

Bieten zum Zeitwert (komplett oder zur Ersatzteilgewinnung)

KRS 4201

Baujahr 1980, Speicher 32 K, mit 4 Magnetbandgeräten ISOT 5003, 4 Trommeln PBB 204-2, Festwechselplatte ISOT 1370

sowie diversen Geräten der 2. und 3. Peripherie.

VEB Schuhfabrik „Banner des Friedens"
Stammbetrieb des VEB Kombinat Schuhe, Bereich ODV
Markwerbener Str. 24
Weißenfels, 4850

Suchen

Ste AFS mit direktverkabeltem Adapter für LW 1.6

Frontblende für LW 1.6

Trägerbaugruppe 1.6 mit Stehbolzen

Bieten im Tausch (z. Zeitw.)

2 Laufwerke 1.2

VEB Geflügelwirtschaft
Bezirk Cottbus Rechenstation
Max-Hannemann-Straße 14
Dahme, 7962
Telefon: 6 79

Wir bieten zum Kauf

- **Diskettenlaufwerke MFS 1.6 mit Zubehör**
- **Farbmonitor für 16-Bit-Rechner Rasterbildschirm**

Wir suchen

BC/PC (8- oder 16-Bit) komplett oder **Rechner, Bildschirm, Tastatur, Drucker** einzeln

VEG Tierzucht Neuenhagen
Am Krankenhaus
Dr. Moeller
Neuenhagen, 1272
Tel. Neuenhagen 4 15, 4 16, 78 08

Wir bieten zum Tausch (Zeitwert)

1 Kassettenmagnetbandeinheit K 5221 mit Kassetten

und

1 Drucker SD 1152 (IFSS)

gegen

Drucker K 63 XX oder EPSON
6 Diskettenlaufwerke MFS 1.6

VEG P Ostrau
Am Güterbahnhof 1
Ostrau, 7303
Telefon: Ostrau 4 73

Tauschen zum Zeitwert:

2 Typenraddrucker SD 1152 mit geteilter Walze Schnittstelle IFSS

Typenraddrucker 1152 einfache Walze Schnittstelle IFSS

gegen

Typenraddrucker SD 1152/Typ 257

bzw. **Nadeldrucker 6314 und Schalterdrucker K 6316**

Hotel Stadt Berlin
Am Alexanderplatz
Koll. Heimer
Berlin, 1020
Telefon: 2 19 41 78

Tauschen zum Zeitwert

Festplatte 3,5", 20 MByte und **Floppy-Disk-LW 8" MF 6400**

gegen **PC 1715 oder BC 5120**

Software-Service Kirchberg
Bahnhofstr. 9, **Kirchberg, 9512**

Tausch zum Zeitwert o. Verkauf

MRES A 5601
6 OPS K 3525
4 OPS K 3526.10
Tastatur K 7634.51

Suchen **PC 1715"**

VEB Bekleidungswerke herdas
Postfach 160
Greiz, 6600

Tauschen zum Zeitwert:

1 Festplatte 20 M (Einbau)
2 1,2-Laufwerke
1 Drucker 1157
Leiterplatten für div. BC
4 Tastaturen 7637
1 3,5-Zoll-Laufwerk mit 20 Disketten
1 MB-Kassettenlaufwerk
1 Rastersichtgerät K 8917
1 Digitalisiergerät HDG K 6401

gegen

10 1,6-Laufwerke, 5,25 Zoll

VEB Technische Gebäudeausrüstung Magdeburg
PSF 89/z. Hd. Leiter ODV
Magdeburg, 3010

Suchen dringend einen **Nadeldrucker** zu kaufen bzw. gegen einen **Thermodrucker TS 80** (zum Zeitwert) zu tauschen.

WTZ Forstwirtschaft Arnstadt
Bahnhofstr. 21
Arnstadt, 5210
Telefon: Arnstadt 81 96

Verkaufen zum Zeitwert

BFA 1720 Rechnungs- und Fakturierautomat
DARO 1372 Datenerfassungsgerät

Suchen **Drucker.**

VEB Thermos Langewiesen
E.-Thälmann-Platz 2
Langewiesen, 6312
Telefon: Ilmenau 23 54

Textverarbeitung

D. Petschke; P. Müntefering: „Textverarbeitung mit Personalcomputern", 1. Aufl. Verlag Die Wirtschaft Berlin 1988, 176 S., 8 Abb., 18 Tab., 11,00 M

Die vorliegende Arbeit reiht sich ein in das Angebot des Verlages, praktikables Informationsmaterial über Standard-Software zum Benutzen von Personalcomputern einem breiten, nicht speziell vorgebildeten Leserkreis verfügbar zu machen – hier auf dem Gebiet der Textmanipulation.

Ein Material wird vorgelegt, welches keine speziellen Datenverarbeitungskenntnisse voraussetzt und das durchaus für einen breiten Leserkreis im Büro- und Verwaltungsbereich Tätigen eingesetzt werden kann. Allerdings habe ich bisher die Reihe zum Personalcomputer 1715 so verstanden, daß sie ein Bausteinsystem bildet, in der keine wesentlichen Wiederholungen auftreten. Das wird aber mit dem 3. Kapitel „Der PC und seine Inbetriebnahme" nicht praktiziert und damit werden 27 wertvolle Seiten „verschenkt", die sinnvoller genutzt wären, das Behandelte durch weitere Schemata und Bildschirm„kopien" zu veranschaulichen, z. B. im 4. Kapitel. Gegebenenfalls erforderliche Teile wären im 2. Kapitel „Wir probieren es gemeinsam. ..." einzubinden, dessen Ausführungen den Anfänger in der Arbeitsplatzrechnernutzung ansprechen wollen.

Eine Bemerkung zum Begriff Textverarbeitung wäre noch zu machen: Es ist streng genommen keine Verarbeitung, sondern reine Zeichenkettenmanipulation, die allein die Form verändert, aber den Inhalt nicht verarbeitet. Sollten wir nicht im Sprachgebrauch exakt sein?

Die Kapitel 4 – Das Textprogramm TP – und 5 – KOMBO-Druck – liefern inhaltlich eine praktikable Übersicht über das verfügbare Funktionenspektrum zur Realisierung der Zeichenkettenmanipulationen einschließlich entsprechender Kommandoübersichten. Im 4. Kapitel werden die Grundfunktionen der Texterstellung, -korrektur, Block- und Dateioperationen, Druckbildgestaltung und die DOT-Kommandos aufgeführt. Kapitel 5 referiert die die Grundfunktionen erweiternden Möglichkeiten, um mehrfache Druckvorgänge zu realisieren, die es auch gestatten, variable Textstellen in vorgefertigten Textschablonen auszufüllen. Hervorzuheben sind dabei die vielfältigen praktischen Hinweise, die erst eine effektive Arbeit mit diesem Textsystem ermöglichen, das ja weitgehend einen Arbeitsplatz für einen Schreibfacharbeiter repräsentiert. Dazu zähle ich u. a. den Abschnitt 5.5. zur Textbausteinnutzung und zur Bearbeitung umfangreicher Dateien. Aus datenverarbeitungstechnologischer Sicht insbesondere wäre es für den Anfänger wichtig, ausführlicher auf das hierarchische Menükonzept einzugehen, um das Verständnis für die Nutzungstechnologie noch besser auszuprägen.

Ein wichtiges, aber wohl zu kurz geratenes Kapitel ist das 6. zur „Organisation der Textverarbeitung", in dem unterstrichen wird, daß in einem Organisationsbereich auch die Anwendung von Textsystemen nicht einfach dem Selbstlauf überlassen werden darf, sondern auch hier der alte Grundsatz der Betriebsorganisation nach wie vor gilt: „Erst die Arbeitsprozesse organisieren und dann darin moderne Informations- und Kommunikationstechnologien nutzen."

Prof. Dr. Reiner Tschirschwitz, Berlin

Informationstechnologie

Wilhelm Steinmüller: „Verdatet und vernetzt". Sozialökologische Handlungsspielräume in der Informationsgesellschaft. Mit einem Vorwort von Hans-Peter Dürr.

Frankfurt a. M.: Fischer Taschenbuch Verlag, 1988, 227 Seiten, mit Resümee sowie ausf. Personen- und Sachregister (Band 3872)

Mit ihrem ersten Buch zum Thema der vielfältigen gesellschaftlichen und sozialökonomischen Wirkungen und Konsequenzen moderner Informations- und Kommunikationstechnologien (IKT) will eine interdisziplinäre Forschergruppe des August-Bebel-Kreises gegen die schönfärberischen Verheißungen von einer postindustriellen „Informationsgesellschaft" und traditionellen technologiepolitischen Denkweisen intervenieren, ohne in einseitige Technikskepsis und -kritik abzugleiten.

Mit ihrer problem- und faktenreichen Publikation treten die Autoren gegen die nach ihrer Ansicht verfehlte Informationstechnologiepolitik in der BRD an und setzen den in gefährliche Sackgassen (irreversible Verdatung, Sozialweltverschmutzung) führenden profitorientierten Leitbildern das von ihnen entwickelte Konzept „sozialökologischer Technologiegestaltung" entgegen. Ihr Analyse- und Konzeptionsgegenstand ist die Praxis aktueller IKT-Anwendungsfelder in der BRD.

Weil die modernen Informationstechnologien um Größenordnungen flexibler und folgenreicher als die bisherigen Maschinentechniken des kapitalistischen Produktionsprozesses sind („Technisierung der Kopfarbeit durchdringt alle gesellschaftlichen Bereiche") sei die Übertragung traditioneller Positionen und Konzepte, insbesondere der Gewerkschaften (Abfederung der technologischen Anwendungsprobleme durch „soziale Gestaltung der Folgen") nicht mehr uneingeschränkt möglich. Daraus erwachse ein erheblicher Diskussions- und Handlungsbedarf zur *politischen* Gestaltung der Informationstechnologien.

Mit ihren Überlegungen appellieren die Autoren vor allem an die Sozialdemokratie und die Gewerkschaftsbewegung, dringend eigenständige, an den Interessen der IKT-Betroffenen orientierte Konzepte und Strategien zu entwickeln und sich bei der Formierung der für einen breiten sozialen Konsens zur gesellschaftskonformen Technologiegestaltung erforderlichen Sammlung gesellschaftlicher Kräfte und Bewegungen an die Spitze zu stellen.

Die Verfasser des Buches „Verdatet und vernetzt" plädieren für eine gründliche und kritische Auseinandersetzung mit den scheinbar objektiven Sachzwängen der modernen IKT, ihren Möglichkeiten, Risiken und Gefahren. Durch die flächendekkende Ausbreitung der Technisierungsprozesse mit Hilfe dieser Technologien auf die Distributionssphäre, den Staat sowie den privaten Wohn- und Freizeitbereich („Industrialisierung des Alltags") entstehen neue Herausforderungen der Interessenallianz und -artikulation aller hiervon betroffenen gesellschaftlichen Gruppen. Im Konflikt zwischen Kapital und Arbeit werden in einer künftigen „informatisierten Gesellschaft" die breiten Schichten des Proletariats (Betroffene) mit modernisierten und vervollkommneten bürokratisch-administrativen Strukturen des Staates und höherer wirtschaftlicher und organisatorischer Steuerungs- und Kontrolleffizienz der Konzerne konfrontiert werden.

Mit ihrer vorgelegten Publikation wollen die Vertreter des August-Bebel-Kreises aktuelle Antworten auf die in der Öffentlichkeit weit verbreiteten Technikzweifel und Befürchtungen geben und Prioritäten für die gegenwärtig unter dem Einfluß der wissenschaftlich-technischen Revolution im Umbruch befindliche kapitalistische Gesellschaft setzen.

Dietrich Gertenbach, Berlin

Mikrorechner Amateurmeßtechnik

A. Bogatz: „Mikrorechner in der Amateurmeßtechnik", Berlin, Militärverlag der DDR (VEB), 1988, 112 S., 96 Bilder, 29 Programme

Ein wichtiges Anwendungsgebiet der Mikrorechner – in Form von Einplatinenrechnern bis zum hochwertigen Kleincomputer – ist ihr Einsatz in Meß- und Steuerungssystemen. Damit werden immer mehr Meßwertverarbeitungsfunktionen vom Bediener weg in das Meßgerät verlagert. Die vorliegende Publikation wendet sich vor allem an Amateure, die an rechnergestützter Meß-, Steuer- und Regelungstechnik interessiert sind. Folgende Themenkomplexe werden behandelt: Standard-Schnittstellen (V.24-, Centronics- und IEC-625-Schnittstelle), Meßsignalerzeugung mit Mikrorechnerausgabegruppen (Erzeugung von Gleichspannungs- und Wechselspannungssignalen), Messung von Gleich- und Wechselspannungen, automatische Meßbereichswahl, Berechnung von Differenzen, Verhältnissen usw., Funktionsgenerator, Frequenzzähler.

Da sich die Broschüre an fortgeschrittene Elektronikamateure wendet, werden Grundkenntnisse digitaler Schaltungstechnik, Grundbegriffe im Umgang mit Mikroprozessoren und deren Peripherie-Schaltkreisen vorausgesetzt. Hervorzuheben ist, daß bei allen Applikationen neben der Hardware auch die Software beschrieben wird.

Klaus Fischer, Berlin

Programmsystem FEPAS/PC

Basierend auf einem bereits in vielen Anwendungsbereichen nachgenutzten Programmsystem FEPAS (Variante OS) wurde von uns eine spezielle PC-Variante entwickelt.
Das Programmsystem FEPAS/PC ist auf 16-Bit-Rechentechnik unter DCP 3.20 lauffähig; einsetzbar auf Hardwarekonfigurationen EC 1834, A 7150 und Kompatible sowie unter SCP 1520.
Die Software ist geeignet zur Analyse folgender Aufgabenklassen:
– elastische Deformation kompressibler Körper (2D-Lösungen)
– Wärmeleitung (2D-Lösungen)
– thermoelastische Deformation kompressibler Körper (2D-Lösungen).
Die zu lösende Programmgröße ist abhängig vom verfügbaren HS des PC (512 KByte: Vernetzung mit etwa 200 Elementen bei 800 Knoten möglich). Die Eingabedatei wird interaktiv erstellt. Nutzerfreundliche Preprozessorfunktionen vereinfachen die Handhabung.
Der Elementkatalog umfaßt isoparametrische 2D-Elementetypen.
Interessenten wenden sich bitte an die TU Karl-Marx-Stadt, Sekt. FPM, WB Theorie der Fertigungs- und Abtrenntechnik, PSF 964, Karl-Marx-Stadt, 9010
Dr. Leopold

uniNet/DOS für PC/AT-Kompatible

Nach der Bereitstellung des Kommunikationssystems uniNet/DOS Version 1.1 zur Einbeziehung von IBM-kompatiblen Rechnern (mit einer COM-Karte) in das Netzwerk der 32-Bit-Rechner SKRnet (edv-aspekte 3/88 und rd 8/88) steht nun auch seit 10/88 uniNet/DOS (FPU) Version 1.1 zur Einbeziehung von PC/AT-kompatiblen Rechnern, die anstelle der asynchronen Kommunikationskarte eine FPU-Karte besitzen, zur Verfügung.
Unterstützt wird standardmäßig die Kommunikation über die Schnittstelle FCOM2, aber auch jede andere verfügbare Schnittstelle ist verwendbar.
Die FPU-Version ist zu einem Aufpreis von 400,– M erhältlich.
Interessenten wenden sich an den VEB Leitzentrum für Anwendungsforschung – Softwarebetrieb CAD/CAM, Betrieb des VE KDV, Jacques-Duclos-Str. 47–52, Berlin, 1156, Tel. 37 80 33 60
Bernhard Genth

Anzeige des Betriebszustandes am A 6402

Zur Überwachung der Systemarbeit des A 6402 wurde eine Schaltung entwickelt, die bereits eine optische Kontrolle ausgewählter Systembus-Signale ermöglicht. Es kann jederzeit ohne Einsatz von Meßtechnik festgestellt werden, ob Bus-Zyklen aktiv oder alle STV-Module und MB-Controller bereit sind.
Die Arbeit des Wartungsdienstes wird durch folgende Besonderheiten erleichtert:
– Abweichungen vom „normalen" Blinkrhythmus der optischen Überwachung können Hinweise für die Fehlersuche sein
– Anordnung auf einem freien Steckplatz, dadurch Überwachung aller dem Prozessor zugewandten STE auf Busfehler
– Durch Stecken einer Minimalkonfiguration und stufenweises Hinzufügen von STE ist eine Fehlereingrenzung schnell möglich.
Der Aufbau erfolgt auf einer Segment-STE mit KBR-typischen Abmessungen (215 × 170 mm). Die Systembus-Belastung entspricht einer Einheits-Buslast je Signal.
Ausführliche Beschreibung und Schaltungsunterlagen stehen zur Nachnutzung der Neuererleistung zur Verfügung.
Interessenten wenden sich bitte an die Deutsche Post, Rundfunk- und Fernsehtechnisches Zentralamt, Agastr. 16, Berlin, 1199, Tel.: 6732321.
Klopp

ARCHIV-Programm

Das Programmsystem ARCHIV ermöglicht die rechnergestützte Verwaltung eines Software-Archivs.
ARCHIV ist in REDABAS und Kompatible geschrieben und dient zur Erfassung, Änderung und Auswertung von Informationen über Programmsysteme und einzelne Files.
Es werden folgende Merkmale gespeichert:
– Name des Programm(-Systems), Version, Erfassungsdatum
– Firma, Abteilung, Autor
– Kurzcharakteristik, Testvermerk
– Programmkenndaten
Softwareklasse
CR-Klassifikation
Rechnertyp, spezielle Peripherie
Betriebssystem, Programmiersprache
HS-Bedarf, Disk-Bedarf
Anzahl der Files
Disketten-Nr.
– Informationen zu den einzelnen Files
Name, Typ, Version, Länge, Check-Summe, Disk-Nr.
Bemerkungen zur Funktion des Files.
Die Angabe aller Merkmale zu einem Prog(-System) ist nicht Pflicht, aber empfehlenswert, denn sie machen das Archiv aussagekräftig und ermöglichen umfangreiche Suchläufe.
Die Angaben zur Softwareklasse, Rechnertyp, Betriebssystem und Programmiersprache werden in einer Deskriptoren-Datei gesammelt. Diese Datei wird erst bei der Arbeit mit ARCHIV nach und nach gefüllt, ihr Inhalt paßt sich daher flexibel an die jeweiligen Bedürfnisse an. Die schon enthaltenen Deskriptoren stehen bei jeder Neuaufnahme zur Auswahl bereit. Durch diese Verfahrensweise wird eine gute Systematik bei der Archivierung erreicht.
Die Arbeit mit ARCHIV erfolgt voll menügesteuert.
Das Suchen von Informationen im Archiv ist mit Hilfe folgender Deskriptoren möglich:
– Für Prog-System-Informationen
Prog-(System)-Name, Version, Softwareklasse, CR-Klassifikation, Rechnertyp; Betriebssystem, Programmiersprache
– Für File-Informationen
Prog-(System)-Name
Version
File-Name
File-Typ.
Bei Angabe mehrerer Deskriptoren gilt die UND-Verknüpfung. Gesuchte Informationen können in einer Auszugsdatei zusammengefaßt, sofort auf dem Bildschirm angezeigt oder ausgedruckt werden.
Interessenten wenden sich bitte an die IH für Seefahrt Warnemünde/Wustrow, Direktorat Wissenschaftlicher Gerätebau/Rechentechnik, WB Informationsverarbeitung, Richard-Wagner-Str. 31, Warnemünde, 2530, Tel.: 57343
Dr. Lübcke

Wir suchen

... eine hardwareseitige Lösung für das Einlesen von 5,25"-Disketten (1.6 LW, beidseitig/doppelte Dichte beschrieben) in den A 6402 mittels Minifolienspeichereinschub K 5661.20.
Erbitten Ihre Hinweise an den VEB Ingenieur-Tief- und Verkehrsbaukombinat Rostock, Dierkower Damm 39, Rostock 40, 2540, Tel.: 393/241
Hamann

Bei der Rechnerkonfiguration PC 1715/LX-86 treten bei der Abarbeitung von PASCAL-Anwenderprogrammen für uns nicht zu erklärende Effekte auf; Drucklisten werden teilweise fehlerhaft ausgegeben. Nur ein Ausschalten des Druckers hilft uns dann weiter! Wer kann in dieser Angelegenheit helfende Unterstützung geben?
Bitte wenden Sie sich dann an die Bäuerliche Handelsgenossenschaft e. G. Auerbach, Rechentechnik, Bahnhofstr. 18, Auerbach, 9700, Tel.: Auerbach 3053
Schwabe

Suchen dringend
PC 1715 möglichst mit Drucker

VEB Kreisbetrieb für Landtechnik
SIB für ZT 300 und Varianten
Meißen Sitz Schletta
Schletta, 8251
Telefon: Meißen 27 72, App. 35

Wir suchen dringend zum Zeitwert
BC A 5120, PC 1715, A 7100
mit Drucker

VEB Institut für Getreideverarbeitung, ORZ
A.-Scheunert-Allee 40-41
Bergholz-Rehbrücke, 1505
Telefon: 2 52, App. 280

Bieten aus Importen
zum Verkauf:

Prüfsimulator für Datenendeinrichtungen (4.019 001.011) Typ: AP. Test

Stückpreis: 12 200,– M

Einsatzgebiet für AP 62/64 (Abonnentenpunkt) in der Datenfernverarbeitung.
Sendegeschwindigkeit 200–9600 Herz,
V. 24-Schnittstelle für Geräteanschlüsse
V. 28-Schnittstelle für Geräteanschlüsse

Bestellungen sind zu richten an:

VEB Robotron-Anlagenbau
Leipzig
Abteilung Ersatzteilexport
Gerberstr. 3
Leipzig
7010

Suchen dringend
Drucker für BC 5120
(auch SD 1152 möglich)

LPG Pflanzenproduktion
Scharfensteiner Str. 54
Großolbersdorf, 9364
Telefon: Wolkenstein 3 42

Suchen zu kaufen
Arbeitsplatzcomputer
AC 7100 oder AC 7150, komplett.

VE Meliorationskombinat Potsdam
Abteilung GW und PM
Schloßstraße 1, PF 43
Potsdam, 1561
Telefon: Potsdam 31 12 74

Suchen 16-Bit-Computertechnik

A 7150
EC 1834
IBM-XT-kompatible
mit Drucker und
Festplatte ≥ 20 MByte,
jedoch nicht Bedingung.

KONSUM-Fleischverarbeitungskombinat Dresden, Sitz Bautzen
Abt. Entwicklung/Überleitung
Wilthener Str. 45a
Bautzen, 8600
Telefon: Bautzen 54 71 65

Suchen dringend

EC 7902 und
FS 1501
(auch zur Ersatzteilgewinnung)

zu kaufen.

Angebote bitte an das
WTZ der Landwirtschaft beim
Rat des Bezirkes Rostock
Eikbomweg 33b
Rostock 5, 2510
Telefon: 84 66, App. 77, 69

Suchen dringend für Steueraufgaben im Versuchsfeld

1 bis 2 MC 80

zu kaufen.

Wir benötigen

Konfiguration
MC 80.3.1.1.1.1.3.

Angebot auch in anderer Zusammenstellung an den

VEB INEX Berlin/BT Leipzig
Käthe-Kollwitz-Str. 1
Leipzig, 7010
Telefon: 3 91 34 12

Wir suchen dringend
Hard- und Softwarelösung für die Daten-ein- und -ausgabe zwischen R 4000 bzw. KRS 4200 und Diskette.

VEB Kraftwerke „Völkerfreundschaft"
Abt. EBR
Hagenwerder, 8905
Telefon: 7/24 91 bzw. 7/25 73

Bieten zum Zeitwert
Datenerfassungskapazität
auf BC A 5120/30
vorwiegend zweite Monatshälfte

VEB Organisation und Rechentechnik
PSF 743 – Zwickauer Str. 296
Karl-Marx-Stadt, 9010
Telefon: 85 22 02, Abt. RS

Suchen
BFA 1720
(auch als Ersatzteilspender)

VEB Energiekombinat Schwerin
Abt. TZ
Obotritenring 40
Schwerin, 2751
Telefon: 8 66 26 58

Verkaufen zum Zeitwert
Beistellgeräte (Floppy-Disk) 8" und
8"-Standarddisketten.

Ostthüringer Molkereikombinat
Industriestraße 1
Gera, 6500
Telefon: 5 12 88

Wir suchen für die Projektierung

ZAK-Nummernkatalog (ZAK-Nummer, vollständige Bezeichnung der Position) für den Bereich der Maschinenbauindustrie auf Datenträger

VEB Blechbearbeitungsmaschinenwerk Gotha
Betriebsrechenzentrum
Mühlhäuser Str. 3
Gotha, 5800
Telefon: Gotha 6 52 19

Wir bieten zum Kauf an:
2 funktionsfähige ZE EC 1040
Baujahr 1977 und 1982
1 GSS EC 5551
6 WPS EC 5052
7 MBG EC 5016
2 LKS EC 7010
2 LKL EC 6012
1 LBST EC 7903
alles Baujahr 1974, zum Zeitwert.

VEB Datenverarbeitungszentrum
Bereich Technik
Leipziger Str. 71
Erfurt, 5024
Telefon: 57 13 54

Verkaufen zum Zeitwert
Multiplexor MPD4 (16 K HS) mit
2 HPK (AS8)
1 HP SAS Beistelltisch
1 SD 1156 Standortperipherie
Baujahr 1978
KRS 4201 (32 K HS) mit HPB (AS3)
und Standortperipherie, Baujahr 1978
3 Magnetbandgeräte EC 5017-02
3 Disketten E/A-Gerät EC 5075

VEB Bandstahlkombinat
„Hermann Matern"
Eisenhüttenkombinat Ost
Werkstr. 1
Eisenhüttenstadt, 1220
Telefon:26 34/31 52/32 24
Telex: 168422-15dd

Wir bieten zum Tausch (Zeitwert)
1 Plotter K 6418
1 Rastersichtgerät K 8917
gegen
5 Diskettenlaufwerke MFS 1.6 (5,25")
mit Wertausgleich

VEB Wasseraufbereitungsanlagen Rathenow
Grünauer Weg 121–123
Rathenow, 1830
Telefon: 6 53 12

Bieten zum Zeitwert
Thermodrucker K 6304
(neuwertig)
Suchen
Nadeldrucker K 6313/6314
oder
Typenraddrucker SD 1152/1156

LPG Pflanzenproduktion Lüdersdorf
Dorfstr. 53
Lüdersdorf, 1301
Telefon: Lüdersdorf 2 08/2 09

Bieten **PC 128 D mit Floppy und Drucker**
zum Zeitwert
suchen
PC 1512 oder 1640 oder Kompatible
(möglichst mit Festplatte)

VEB Limona Weimar, Abt. ORZ
Schützengasse 9
Weimar, 5300

Tauschen z. Zeitw. (ggf. auch Verkauf)
1 Seriendrucker 1154 gegen
1 Drucker vom Typ
1152; 1157; FX 1000; 63xx

VEB ITVK Rostock, ORZ
Dierkower Damm 39
Rostock, 2500
Telefon: 3 93–2 41/–2 35

Bieten zum Zeitwert
2 Commodore 128 D (Personalcomputer) jeweils mit Drucker, Monitor 80 × 24, kompl. Dokumentation und Software
zum Tausch gegen
1 BC 5120/30 oder K 8924 (evtl. auch PC 1715, AC 71xx) jeweils mit Drucker, MFS 1.6 und Monitor 80 × 24.

VEB Zuckerfabrik „Fritz Reuter", Abt. ORA
Bluthsluster Str. 24
Anklam, 2140

Tauschen
z. Z. infolge Typenbereinigung
A 5120 mit MON 2
3 1.6 Laufwerke, Tastatur K 7636 und Drucker 1152
gegen
PC 1715 mit 2.1.6. Laufwerke und Drucker
Suchen
1 MFS 6400 für PC 1715 mit 108" Disketten.

Landwirtschaftliche Produktionsgenossenschaft „Ernst Thälmann"
Manker, Kr. Neuruppin, 1901

Wir bieten zum Zeitwert
1 Plotter A3/A4 (HP-kompatibel)
4 Magnetbandschränke PT-3M
2 Klimatruhen KT2.1
Wir suchen
AC 7150 G und **PC 1715**
mit Wertausgleich

VEB BMK Kohle und Energie
KB Fopro Berlin
Am Köllnischen Park 1, PSF 9
Berlin, 1020

Suche Bedienendrucker BD 4000
Biete **Magnettrommelspeicher PBB 204/2** von Elwro
zum Zeitwert

VEB Kalibetrieb Zielitz
DB Fabrik/Prozeßrechner
Zielitz, 3215
Telefon: Loitsche 40/28 77

Bieten zum Zeitwert
Drucker SD 1152
Interface IFSS, Gummiwalze
Suchen mit Wertausgleich
Drucker K 6313, V 24

LPG Pflanzenproduktion Hennickendorf
Sitz: **Ruhlsdorf, 1711**

Bieten zum Zeitwert
2 Rastersichtgeräte K 8917
2 Laufwerke 8" EC 5074
1 A 5130
Suchen
1 PC 1512 oder PC 1640 oder A 7150
1 Laufwerk 1.6, 5,25"

VEB Ziegelwerke Halle
Ingenieurbüro Bau- und Grobkeramik
Burgstr. 38, Postfach 99
Halle, 4020

Wir bieten zum Zeitwert
2 Plotter K6418
1 HDG K6401
Wir suchen
PC 1715
mit Wertausgleich

VEB BMK Kohle und Energie
KB Fopro Berlin
Am Köllnischen Park 1
PSF 9
Berlin, 1020

Wir bieten (Zeitwert)
zum Tausch
Laufwerk Original Schneider PC (Amstrad) 48 tpi, 360 K
Suchen
Co-Prozessor 8087, 8 MHz

VEB Spezialbaukombinat Wasserbau
Forschungsstelle Blankenburg
Grefestraße
Blankenburg/Harz, 3720
Telefon: 7 44 20

Suchen dringend
1 Schneideautomat Typ „Fimafold"
1 Bildschirmsteuergerät EC 7922.01 M
Bieten zum Zeitwert
EC 1022 folgender Konfiguration
Baujahr 1977
1 Zentraleinheit EC 2622
1 AE EC 7077 mit Datensichtgerät
1 GSS EC 5551
8 WPS EC 5052
1 MBS EC 5517
5 MBG EC 5017
1 PD EC 7033

VEB Eisengießerei u. Maschinenfabrik ZEMAG Zeitz
Schädestraße, **Zeitz, 4900**
Abt. OR, Telefon: 8 54 15

Verkaufen zum Zeitwert
1 LBE K 6200, neuwertig
1 LBL 1210-332, neuwertig
5 MFS 1.2-Diskettenlaufwerke (Zeitwert)
evtl. Tausch gegen
Drucker K 63 xx

ZBO Landbau Erfurt
– Rechenstelle –
Spielbergtor 12d
Erfurt, 5084
Telefon: 5 12 11

Großformatige Schrift für PC 1715

Das Programm zur Erzeugung großformatiger Schrift entstand Mitte des Jahres unter dem Aspekt der optischen Auswertung von Dokumentationen. Es wird derzeit u. a. zur Beschriftung von Haftetiketten verwendet.

Das Programm ist in PASCAL geschrieben und läuft auf PC 1715/SCP. Außer einer Abfrage der Druckerbereitschaft werden keine Systemaufrufe benutzt. Derzeit sind zwei Zeichensätze realisiert, die entsprechend der Möglichkeiten des zum Test verwendeten Druckers K 6313 modifiziert werden können: Druck im Grafikmodus normaler oder doppelter Dichte (breit/schmal), fetter oder lichter Abdruck (voll/hell), normaler oder inverser Abdruck.

Die Schriftart *Gotic* ermöglicht Groß- und Kleinschreibung, ansonsten werden Klein- in Großbuchstaben umgewandelt, nicht im Zeichensatz enthaltene Eingaben werden als Leerzeichen dargestellt. Zur Erleichterung der Erstellung der Zeichensätze entsteht derzeit ein Editor. Weitergehende Funktionen der Zeichenbehandlung wie Zoom usw. sind nicht vorgesehen und erscheinen aufgrund der breiten Palette an 16-Bit-Grafikprogrammen auch sinnlos.

Auf Anregung vom Softwarezentrum Sömmerda wird ein Textmodus realisiert, der das Ausdrucken von Textdateien mit Steuerzeichen ermöglicht, die mit TP erstellt wurden. Je nach gewählter Schriftart wird die maximale Zeichenzahl pro Zeile ermittelt; überzählige Zeichen werden abgeschnitten.

Das eigentliche Programm ist sehr klein, den meisten Platz belegen die Bitmuster für die Schrift. Um mit dem ohnehin knappen Speicherplatz auszukommen, wurden die Zeichensätze als Overlay gehalten; nächster Schritt sind nachladbare und beliebig editierbare Dateien. Die Bedienung des Programms erfolgt menügesteuert (Funktionswahl durch Cursortasten).

An eine wesentliche Erweiterung des Programms ist aus den schon genannten Gründen und der steigenden Verfügbarkeit von 16-Bit-PC mit entsprechender Software nicht gedacht.

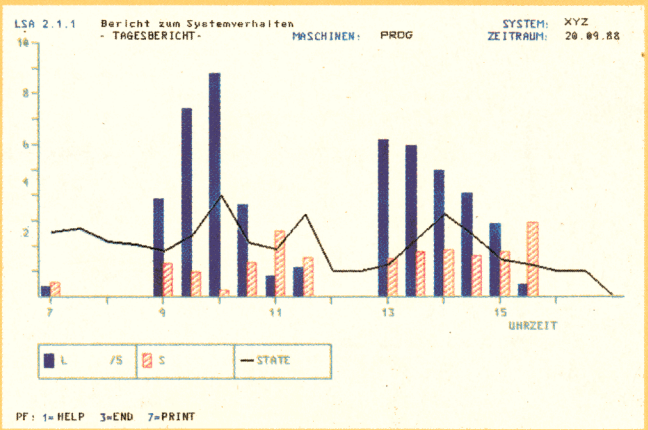
Frank Bartnitzki
VEB Bekleidungswerk Erfurt
Stammbetrieb, ORZ,
Anger 33, Erfurt, 5020

83517 3 144 975 901
FORSTWIRTSCHAFT
7002 0147 PSF 9

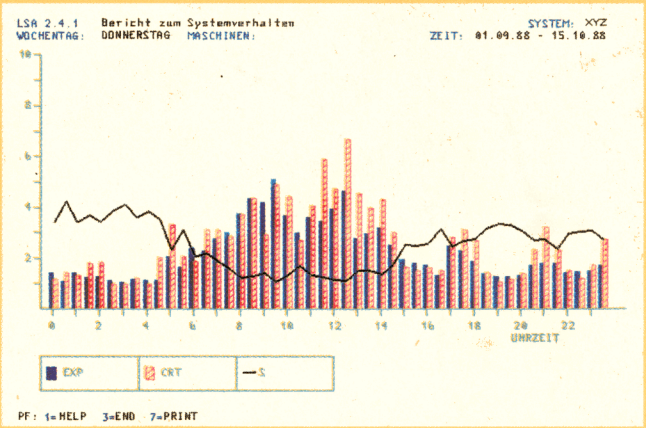
software

LS2 Softwaremonitor

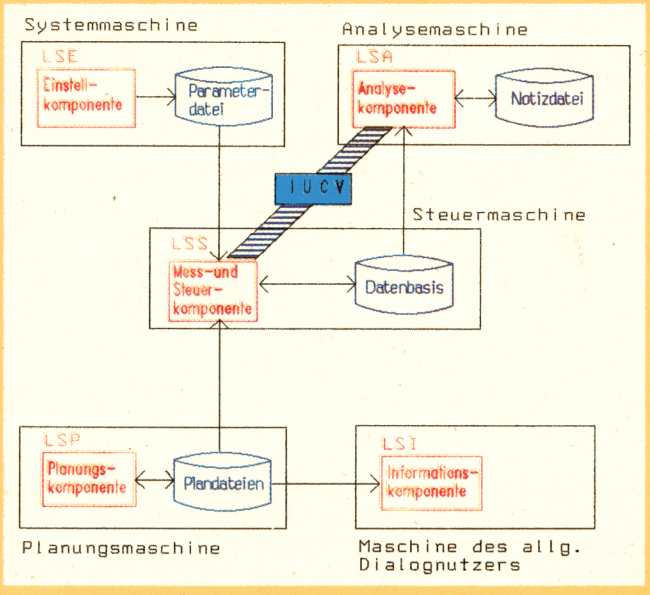
LS2 ist ein Softwaremonitor mit zusätzlichen Steuerungsfunktionen für SVM. Als Monitor erlaubt LS2 sowohl die operative Anzeige von Meßwerten als auch postoperative Analysen nach verschiedenen Gesichtspunkten und mit verschiedenen Formen der Ergebnisdarstellung. Im Beitrag auf Seite 19 dieser rd-Ausgabe, dem auch die abgebildeten grafischen Darstellungen auf dieser Seite entnommen wurden, wird LS2 ausführlich beschrieben.



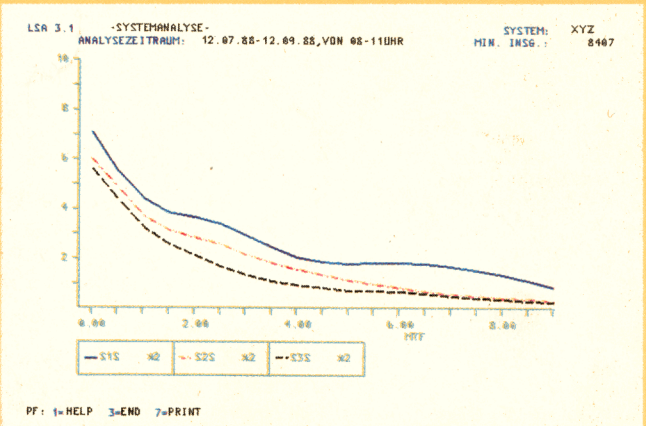
Tagesbericht für eine Nutzergruppe



Wochentagsanalyse von Servicekenngrößen zum Gesamtsystem



Die Softwarestruktur von LS2



Service für die einzelnen Servicegruppen in Abhängigkeit von der Last

ISSN DDR 0300-3450 · rechentechnik/datenverarbeitung · Berlin 26 (1989) 3 · S. 1–40